# A Morbit problem to solve

A quote by Melinda Chapman has been encoded using the Morbit Cipher for you to decode. You are told that **8**=**××**, **5**=**●●**, **9**=**×●**, **4**=**●×**, **2**=**●–**, **6**=**–×**

5 8 5 5 9 6 5 2 9 8 2 8 5 4 5 5 3 7 3 2 4 1 6 5 3 4

7 4 3 6 4 7 3 7 9 1 3 2 6 9 4 9 1 9 3 6 4 7 3 5 4

4 2 4 9 9 5 6 4 2 4 1 7 3 9 5 4 5 3 4 7 4

# Background on Solving Morbit

The Morbit cipher works by first converting the text into Morse code which is written as a series of dots (●), dashes (–), and spaces. To make it more convenient to solve, we typically represent the spaces as an ×. A single space is used at the end of a Morse code letter and a pair of spaces is used at the end of a word.

The person encoding the text then decides which of the digits 1-9 will stand for each of the 9 combinations of dots/dashes/spaces with no restriction on that choice. For example, ×● could be represented by 4 and ×× represented by 9. Given the mapping of the digits, the Morse code is translated to the cipher text by taking the pairs of morse code digits and finding the corresponding cipher text mapping..

Decoding a Morbit applies the process in reverse. It starts by mapping the known digits to their corresponding dot/dash/space pairs and looking for complete Morse code characters. A complete Morse code character is one where an uninterrupted series of dots/dashes are delimited by a space. For example: ●●●× at the beginning represents the very familiar letter S (three dots). Finding ×●●× in the middle would represent the letter I (two dots). However, if we had ×● × (with an unmapped digit after the dot), we wouldn’t know what the plain text is until we figured out the mapping for the digit although it is either going to be E at the end of a word if the space maps to ×, I if it maps to ● or A if it maps to –.

With that in mind, the strategy for solving a Morbit consists of a set of steps:

1. Build a table of the possibilities for the digits.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |

1. Fill in the table with the known mappings and then just put the remaining mappings in for everything else since we don’t know what they map to.
2. Underneath the digits of the cipher, fill in the known mappings with the corresponding Morse code character (●, –, ×).
3. Solve.. As digits are eliminated, removed them from the possibility table and fill in known mappings under the cipher text. One special case that makes it easier to solve. If you eliminate × as a possibility, leaving ● or –, filling in the corresponding cipher spot with ? makes it easier to find places where a × belongs.

Some good solving rules that help quickly solve a Morbit

1. The first character will never be an ×. If the cipher digit at the start could map to a sequence starting with an ×, you can eliminate that choice.
2. There will never be three spaces (×××) in a row. Hence if you find a cipher digit that is doubled, you know that it can’t map to a ×.
3. Also looking for three spaces, if you have an × right next to an unknown then you know that the unknown can not be ××.
4. No Morse letter is more than 4 dots/dashes and all numbers are exactly 5 dots/dashes. If there is a sequence of 6 characters with an unknown and all the remainder are known to be a dot/dash (●–?) then you know that the unknown must be a ×.
5. If you have four dots/dashes in a row next to an unknown, then the unknown must have an × in it.
6. Not all sequences of 4 dots/dashes are legal Morse characters. (●●––, ●–●–, –––●, and ––––). If you have a pattern that would map to it, you know that you can eliminate it.

# How to solve

Since we are told the mapping of 859426 ciphertext, we can build the following table noting that –●, –– and ×– are unused so we can put them in as options for 1, 3 and 7:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| –● –– ×– | ●– | –● –– ×– | ●× | ●● | –× | –● –– ×– | ×× | ×● |

Based on that information we can map the cipher text as:

5 8 5 5 9 6 5 2 9 8 2 8 5 4 5 5 3 7 3 2 4 1 6 5 3 4

●●××●●●●×●–×●●●–×●××●–××●●●×●●●● ? ? ?●–●× ?–×●● ?●×

I / H A V E/ A / S

7 4 3 6 4 7 3 7 9 1 3 2 6 9 4 9 1 9 3 6 4 7 3 5 4

?●× ?–×●× ? ? ?×● ? ?●––××●●××● ?×● ?–×●× ? ?●●●×

E / I / E

4 2 4 9 9 5 6 4 2 4 1 7 3 9 5 4 5 3 4 7 4

●×●–●××●×●●●–×●×●–●× ? ? ?×●●●●×●● ?●× ?●×

E R / E V E R H

At this point in time, three ciphertext characters still need to be mapped. Looking for spans of at least four ●s and –s next to an unknown, we see the Morse sequence ●●●● followed by the unknown 3 which can only map to ×– since that is the only option which has an × in it. So we mark 3 as ×– and eliminate that from 1 and 7.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| –● –– | ●– | ×– | ●× | ●● | –× | –● –– | ×× | ×● |

Based on that information we can map the cipher text as:

5 8 5 5 9 6 5 2 9 8 2 8 5 4 5 5 3 7 3 2 4 1 6 5 3 4

●●××●●●●×●–×●●●–×●××●–××●●●×●●●●×––?×–●–●×–?–×●●×–●×

I / H A V E/ A / S H C I N

7 4 3 6 4 7 3 7 9 1 3 2 6 9 4 9 1 9 3 6 4 7 3 5 4

–?●××––×●×–?×––?×●–?×–●––××●●××●–?×●×––×●×–?×–●●●×

/ M E Y / I / E M E B

4 2 4 9 9 5 6 4 2 4 1 7 3 9 5 4 5 3 4 7 4

●×●–●××●×●●●–×●×●–●×–?–?×–×●●●●×●●×–●×–?●×

E R / E V E R T H I N

This leaves us with only two options to try. Looking at the 373 we have the sequence ×––?×. If we let 7 be –● then it would map to the letter G giving us SHGC, but if we map it as –– it reads SHOC which makes much more sense, so we know that 7 must be –– and that leaves –● for 1. This results in a mapping table of.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| –● | ●– | ×– | ●× | ●● | –× | –– | ×× | ×● |

Based on that information we can map the cipher text as:

5 8 5 5 9 6 5 2 9 8 2 8 5 4 5 5 3 7 3 2 4 1 6 5 3 4

●●××●●●●×●–×●●●–×●××●–××●●●×●●●●×–––×–●–●×–●–×●●×–●×

I / H A V E/ A / S H O C K I N

7 4 3 6 4 7 3 7 9 1 3 2 6 9 4 9 1 9 3 6 4 7 3 5 4

––●××––×●×––×–––×●–●×–●––××●●××●–●×●×––×●×––×–●●●×

G / M E M O R Y / I / R E M E M B

4 2 4 9 9 5 6 4 2 4 1 7 3 9 5 4 5 3 4 7 4

●×●–●××●×●●●–×●×●–●×–●––×–×●●●●×●●×–●×––●×

E R / E V E R Y T H I N G

Now that we have mapped all the ciphertext characters, the decoded Morse code is the answer:

I HAVE A SHOCKING MEMORY I REMEMBER EVERYTHING