

# Post Correspondence Problem (Domino problem)

Shaun Schreiber  
16715128

9 February 2014

The domino problem is an example of a problem where it is impossible to construct a single algorithm that leads to a correct match-or-no match answer. Although the domino problem can be explained in a number of ways the simplest involves the use of tiles. Thus for the purpose of this document we will define it as a set of tiles where each tile is divided into two sections(top and bottom). Each section contains a set of characters. This can be seen by the visual representation below:

$$\left\{ \begin{bmatrix} i \\ ep \end{bmatrix}, \begin{bmatrix} p \\ pi \end{bmatrix}, \begin{bmatrix} ep \\ p \end{bmatrix}, \begin{bmatrix} pie \\ e \end{bmatrix} \right\}$$

The goal is to order these tiles so that the string that is formed by all of the top sections is equal to the string that is formed by all of the bottom sections. This is known as a match (Note repetition of a tile is permitted). See visual representation below for a match:

$$\left\{ \begin{bmatrix} p \\ pi \end{bmatrix}, \begin{bmatrix} i \\ ep \end{bmatrix}, \begin{bmatrix} ep \\ p \end{bmatrix}, \begin{bmatrix} p \\ pi \end{bmatrix}, \begin{bmatrix} pie \\ e \end{bmatrix} \right\}$$

Getting a match isn't always possible. If you look at the following set of tiles it can never have a match because the top and bottom doesn't even consist of the same characters.

$$\begin{bmatrix} i \\ p \end{bmatrix}, \begin{bmatrix} p \\ i \end{bmatrix}, \begin{bmatrix} d \\ z \end{bmatrix}$$

Now that you know what a match is and that there are sets of tiles that do not have a match. Can we then now find an algorithm that can determine whether any set of tiles have a match? The answer to that question is no. Lets look at the following example.

$$\begin{bmatrix} i \\ p \end{bmatrix}, \begin{bmatrix} p \\ i \end{bmatrix}, \begin{bmatrix} i \\ i \end{bmatrix}$$

This example does not have a match. For a computer to determine this it will have to go through all of the possible combinations which there are infinitely

many because we are allowed to use a tile more than once. This is thus not solvable using the first algorithm. This is why the domino problem is an example of a problem where it is impossible to construct a single algorithm that leads to a correct match-or-no match answer.

## References

- [1] Sipser, M. Introduction to the Theory of Computation. Cengage Learning. 2013.