

Programming Assignment 7

Arindam Karmakar

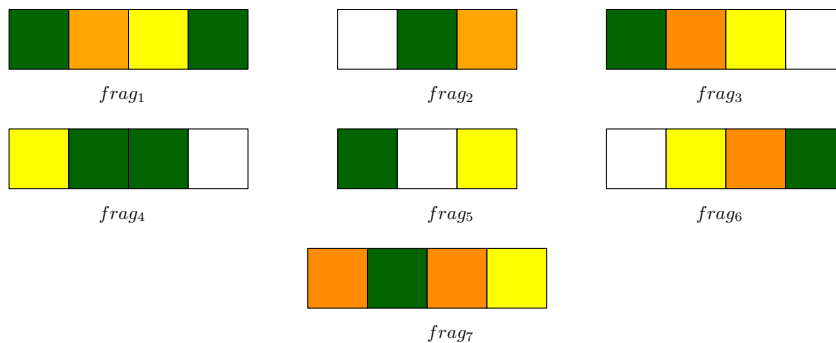
0.1 Write C/C++/Python code for the following problems:

Suppose you have a box of multicolored ribbons in your house. All the ribbons are of the SAME design and color. Your niece and nephews visit you in the summer vacation and got hold of the ribbon box. They also somehow managed some scissors and started the classic activity of kids - paper-cutting. They randomly cut the ribbons into multiple fragments of various sizes (as shown in the figure below). To stop the children trashing your home further, you offer them a pizza if they can create ONE original ribbon from the fragments they have cut. The figure in the next page shows how the fragments can be assembled to form the original one. In the world of computer science these problem can be stated as:

Problem definition

Given a set of fragments $F = \{frag_1, frag_2, \dots, frag_n\}$, find the MINIMUM length ribbon which contains all the fragments. The fragments have the following properties:

- fragments can be considered as strings where each character represents a colour, e.g, $frag_1 = \text{"GOYG"}$
- each fragment $frag_i$ overlaps with ATLEAST one fragment $frag_j, i \neq j$
- no fragment $frag_i$ completely overlaps/contains any other fragment $frag_j, i \neq j$
- union of all the fragments is the original ribbon



The fragments can be assembled in the following way to generate the output (as shown in the Figure 1).

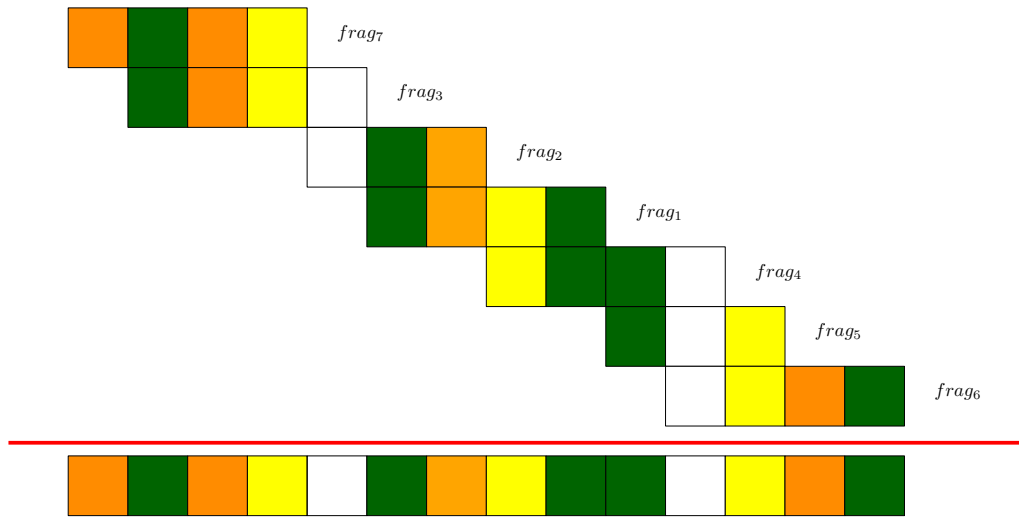


Figure 1: Output Ribbon (original)

After solving the 1-dimensional version, try to solve the 2-dimensional version of the same problem as in the following Figure 2. In this case we have broken fragments $TF = \{tf_1, tf_2, \dots, tf_n\}$ of same styled tiles. In this problem we have to find a way to recover the original tile from the fragments. explain the time complexity of both the algorithms (2d & 3d versions).

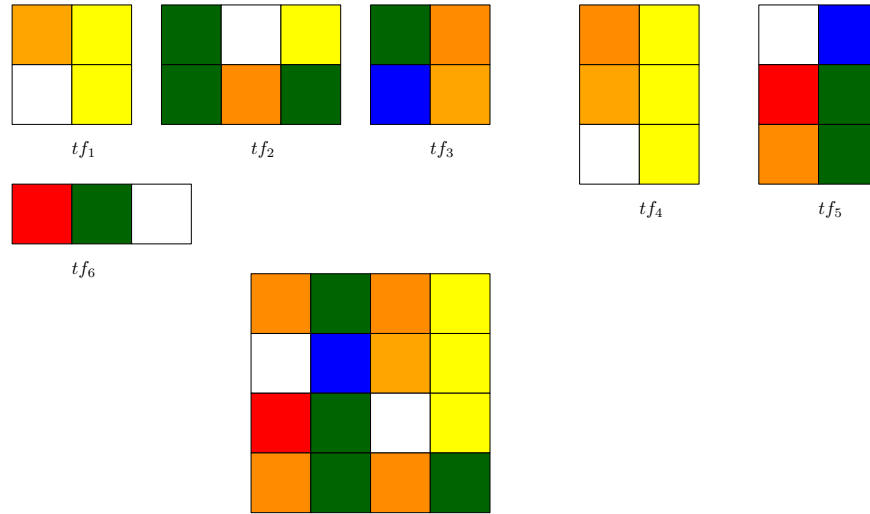


Figure 2: Output Tile (original)