In this problem, we used the partial code a lot.

The most important detail we need to use is listed following.

1.Layout.h && Layout.cpp

class Layout, for this, we overload operator>> to let it can be directed read from input files.

2.GraphPacking.h

we found that it is used by class PackingCommand, and we don't need to use it by ourself. But to fast the calculating, we change the method which use SP to locate rectangles, now the time complexity is about O(n^2+n^2+2n) instead of O(n^2+n^3).

3.PackingCommand.h && PackingCommand.cpp

we need to record and set the status for better results, so we add getS() and setS() function. we make them real virtual and we realize they in SPPackingCommand.h as inline function. We change next() to let it can realize the three operations mentioned in the IV part of the paper.

4.SPPackingStrategy.h

According to the paper, we change initialPacking(), make the initial SPs same instead of both random, in order to correspond to a linear horizontal arrangement of modules.

5.test.h

this provides a method to find out overlap, so can test out results.

6.main.cpp

in this file, we use simulated annealing method, the temperature is rely on the rectangles' number, the factor is infected by the average size of rectangles.