

## Exercises for chapter: Line breaking

1. In dynamic programming, many solutions start from a final stage and work backwards. Why is this approach inappropriate for  $\text{\TeX}$ 's line breaking algorithm? Why would it be even less appropriate for a page breaking algorithm?
2. Not every word should be added to the active list. For instance, for any realistic line length, the second word in the paragraph will not have a valid breakpoint after it, so we need not consider it. Take the model implementation and add this modification. Measure the improvement in running time, for instance by counting the number of calls to some inner routine. Give a theoretical argument for how this reduces the complexity of the algorithm.
3. The total fit code does not yet contain the equivalent of  $\text{\TeX}$ 's `\adjdemerits`. Add that.
4. Add the functionality for hanging indentation to this code.
5. (bonus point exercise)  $\text{\TeX}$  has the possibility of forcing a paragraph to be a line longer or shorter than optimal. Implement that.