

## Efficient Parallel Algorithms - Problem Sheet 4

Consider the problem of parenthesis matching, i.e. finding opening and closing parenthesis pairs in a (long enough) sequence of the kind '((( )))'.

1. Give a formal recursive definition for a valid sequence of parentheses.
2. Give a BSP algorithm that determines whether a given sequence of parentheses is valid, which means that for every opening '(' there is exactly one closing ')' and vice versa.
3. Extend this algorithm to output the matching pairs of '('s and ')'s.
4. We now allow two kinds of parentheses, say '(' , ')', '[' and ']'. Give a BSP algorithm that determines whether this sequence is valid and outputs the matching pairs.

Give asymptotic estimates on computational complexity, necessary communication and required memory for all the algorithms.