

```

function parsetree_by_index(tree){
...counter++;
...var end_parse = false;
...// console.log('switch to: ' + counter)
...// console.log(tree);
...var message = '';
...switch (counter){
...case 1:
...message = 'delete spaces';
...// console.clear();
...var temp = tree.leaf.content;
...//
...https://stackoverflow.com/questions/4025482/cant-escape-the-backslash-with-regex#4025505
...// http://www.javascripter.net/faq/backslashinregularexpressions.htm
...tree.leaf.content = temp.replace(/\\s/g, '');
...break;
...case 2:
...message = 'parse brackets';
...result = parse_brackets(tree);
...break;
...case 3:
...message = 'parse plusminus';
...result = remove_operators(tree, 'plusminus');
...break;
...case 4:
...message = 'parse timesdivided';
...result = remove_operators(tree, 'timesdivided');
...break;
...case 5:
...message = 'unify subscript and exponent (part 1)';
...unify_sub_exponent(tree);
...break;
...case 6:
...message = 'parse integral';
...parse_integral(tree);
...break;
...case 7:
...message = 'parse square root / nth root';
...parse_nthroot(tree);
...parse_sqrt(tree);
...break;
...case 8:
...message = 'parse log_base';
...parse_log_lim(tree, 'log'); //log
...// check_children(tree);
...break;
...case 9:
...message = 'parse lim';
...parse_log_lim(tree, 'lim'); //lim
...// check_children(tree);
...break;
...case 10:
...message = 'parse functions';
...parse_function(tree);
...// check_children(tree);
...break;
...case 11:
...message = 'parse fractions';

```

```

.....parse_frac_textcolor(tree, 'frac');
.....break;
.....case 12:
.....message = 'parse_textcolor (unit)';
.....parse_frac_textcolor(tree, 'textcolor');
.....break;
.....case 13:
.....message = 'delete single \xA7 nodes'
.....var list_of_free = delete_single_nodes(tree);
.....break;
.....case 14:
.....message = 'parse_greek';
.....parse_greek(tree);
.....break;
.....case 15:
.....message = 'parse_numbers';
.....parse_numbers(tree);
.....break;
.....case 16:
.....message = 'delete single \xA7 nodes'
.....var list_of_free = delete_single_nodes(tree);
.....break;
.....case 17:
.....message = 'unify_subscript (part 2)';
.....unify_sub_or_power(tree, false);
.....break;
.....case 18:
.....message = 'parse_subscript'
.....parse_sub_power(tree, false);
.....break;
.....case 19:
.....message = 'unify_power (part 2)';
.....unify_sub_or_power(tree, true);
.....break;
.....case 20:
.....message = 'parse_power'
.....parse_sub_power(tree, true);
.....break;
.....case 21:
.....message = 'delete single \xA7 nodes'
.....var list_of_free = delete_single_nodes(tree);
.....break;
.....case 22:
.....message = 'parse_factors';
.....parse_factors(tree);
.....break;
.....case 23:
.....message = 'delete single \xA7 nodes';
.....var list_of_free = delete_single_nodes(tree);
.....break;
.....default:
.....message = 'end of parse';
.....end_parse = true;
...}
...// check_children(tree);
...return [message, end_parse];
}

```