Wahrheit und method

Sebastian Michoń

1 On The Unmaker, I: elements

1. [0, button, color1, color2] - paints button from color1 to color2.
2. [1, button, innerHTML1, innerHTML2] - changes innerHTML of a button from innerHTML1 to innerHTML2.
3. [2, list, element] - appends element to a list.
4. [3, 'variable_name', value1, value2] - changes this.variable_name from value1 to value2.
5. [5, fun, inv_fun, args] - execute fun on arguments args; in order to reverse changes, uses inv_fun on same arguments.

2 On The Unmaker, II: Inner works

1. Unmaker reverts changes done by state maker, that ought to be reversed (examples of changes, that shall not be reversed are variables changed only once and used only since then or innerHTML that may be constant during whole execution).
2. Unmaker can paint, pop from list, change value or innerHTML to previous value or execute certain function with given arguments. It executes in reversed order compared to StateMaker.
3. StateMaker has array staat, in which changes to "this" that ought to be reversed and changes to buttons are kept in format described above. The values of "this" variables don't change until the end of a function.

3 On The Maker and Unmaker, III: Use

1. Unmaker actions depend on clicking "Previous" button, they're coded in Temp.js, so one does not have to do anything in code of an algorithm.
2. Maker has to execute at its end this.transformator(staat) (staat is just a convention).

4 Primitive Root

1.

5 NTT/FFT animation: states

1. Writing indexes and values of both polynomials to multiply.
2. Formulate either primitive root (NTT) or starting point (FFT)

3. Write first two roots (1 and either function of proot - NTT or starting point - FFT)
4. Construct butterfly
5. Rewrite sequence according to butterfly indices
6. First part of double - calculate the left part of next polynomial - $A_{i,x,2k}$
7. Second part of double - calculate the right part of next polynomial - $A_{i,x,2k+1}$
8. Repeated values of polynomials in points (roots of unity)
9. Multiplication of polynomials
10. Showing inverse roots
11. Ending (After interpolating C(x))- multiply by inversion