Trie

What is Trie data structure?

known as a prefix tree, that is used to store a dynamic set of strings where the keys usually represent sequences of characters. This comes from the word "retrieval" because it provides a way to quickly retrieve data based on keys.

Why we learn this ds?

Gain Deeper understanding on tree ds.

Efficiency in string operations

Auto completion and searching algorithom.

Data compression techniques

Networking and routings

Data Storage and Retrieval

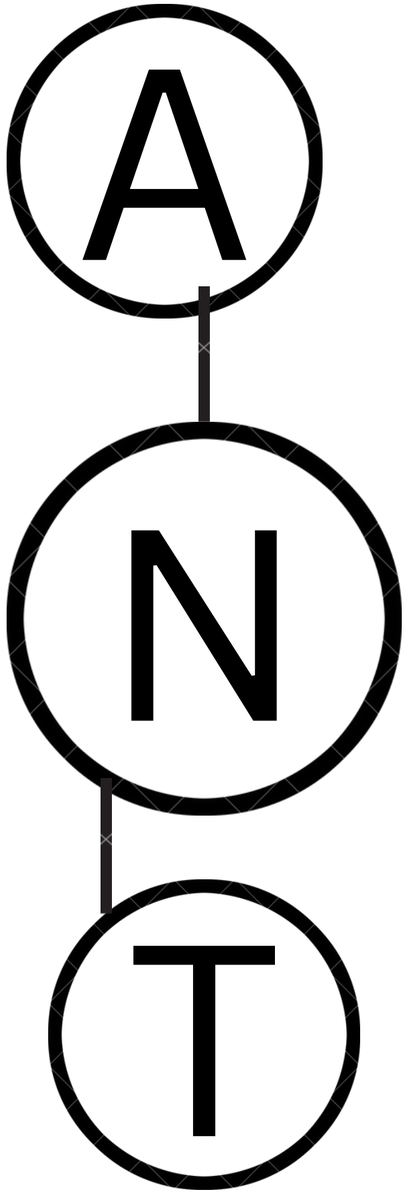
Properties of trie ds

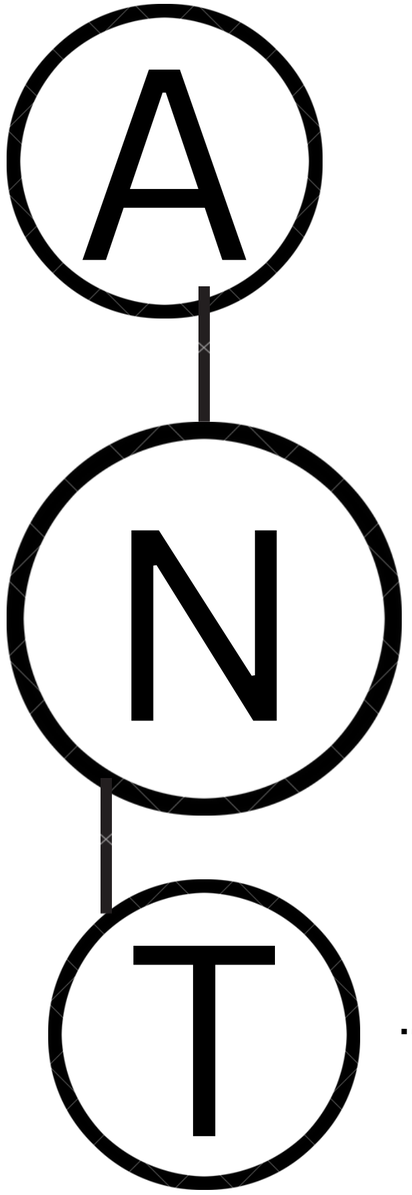
Node structures

- character
- end of word
- connected by edge.

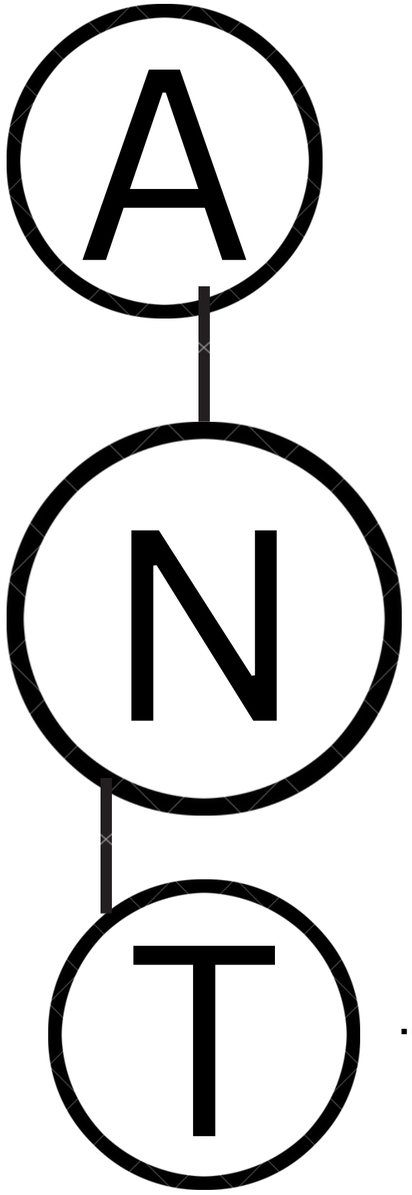
root node is always empty

Child nodes :-
represent the next character of it's parent node.


A
N
T



end of the word

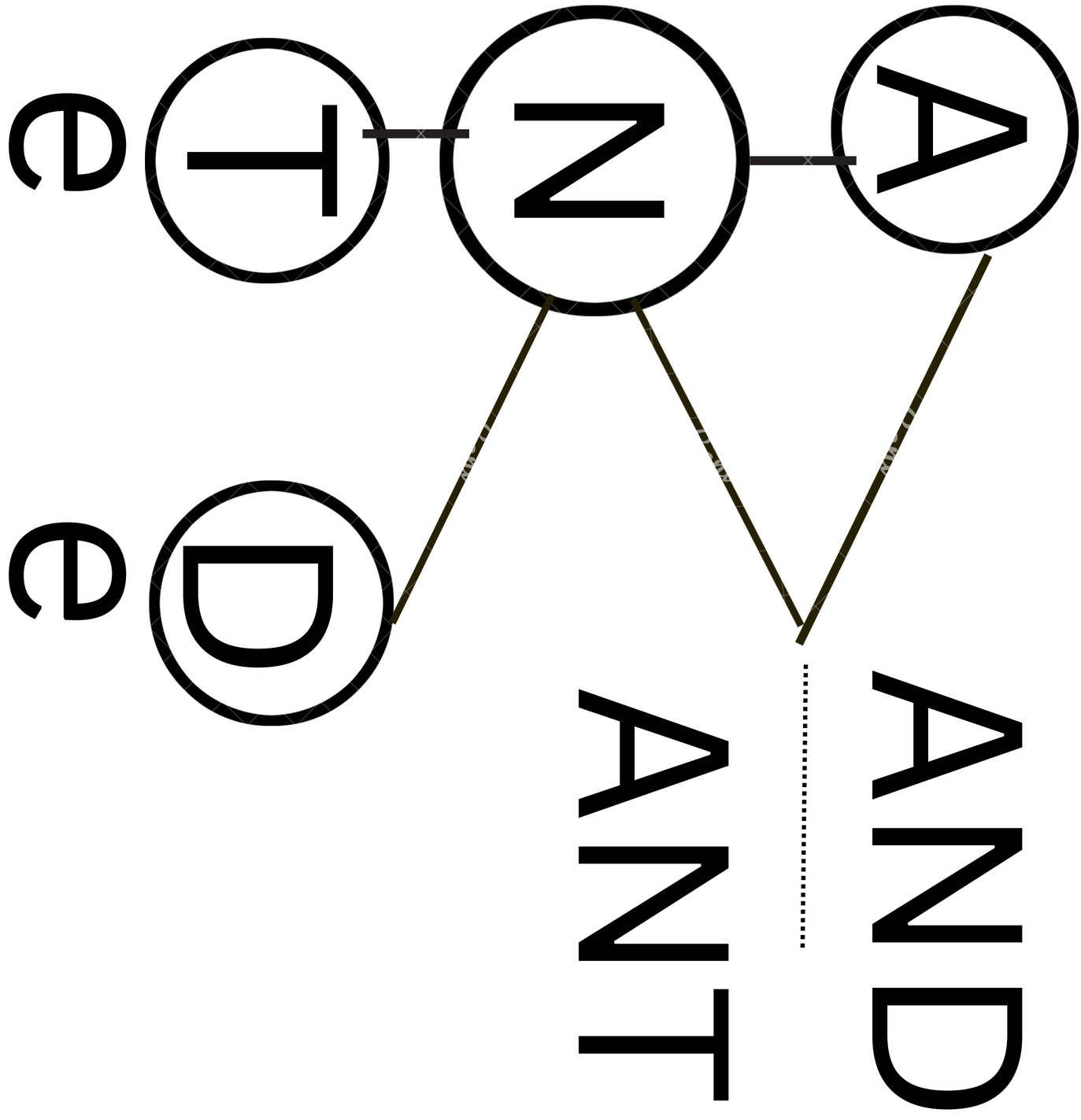


when find a character end of a word then cosider a word otherwise each node contains as a prefix node.

.....

end of the word

Here A and N are prefix node



every 1st char is added with root node.

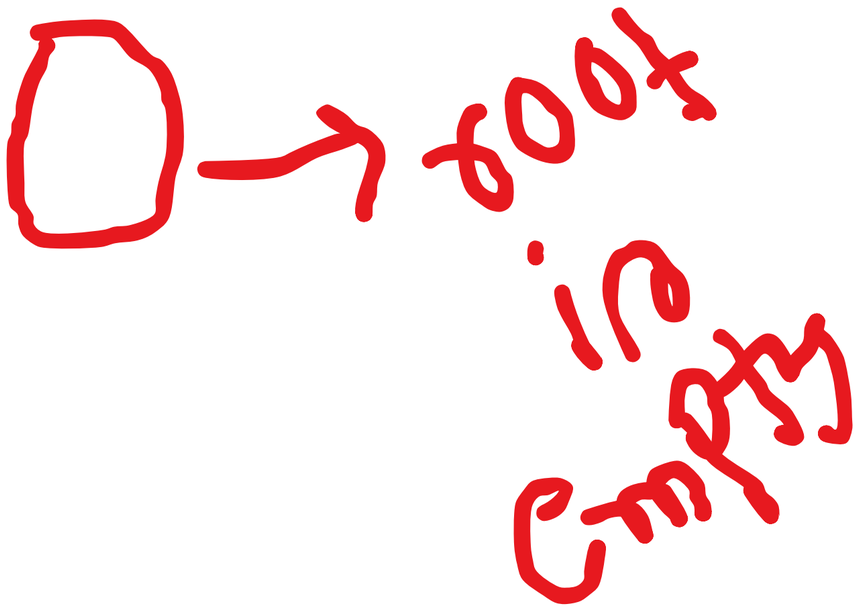
AND

ANT

Now, see an example of insertion at here.

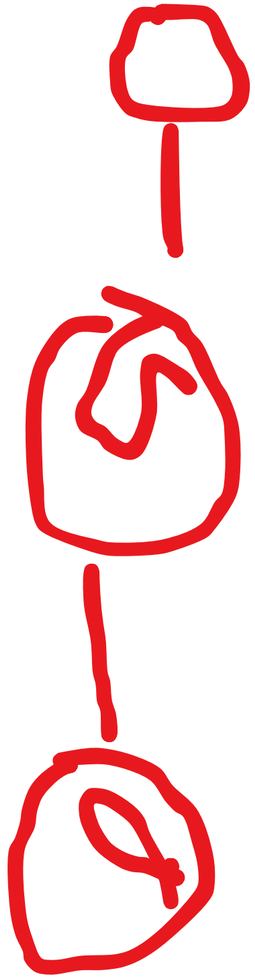
- algorithom
- algo3370
- algo
- like
- life
- amit
- arpon

algorithom

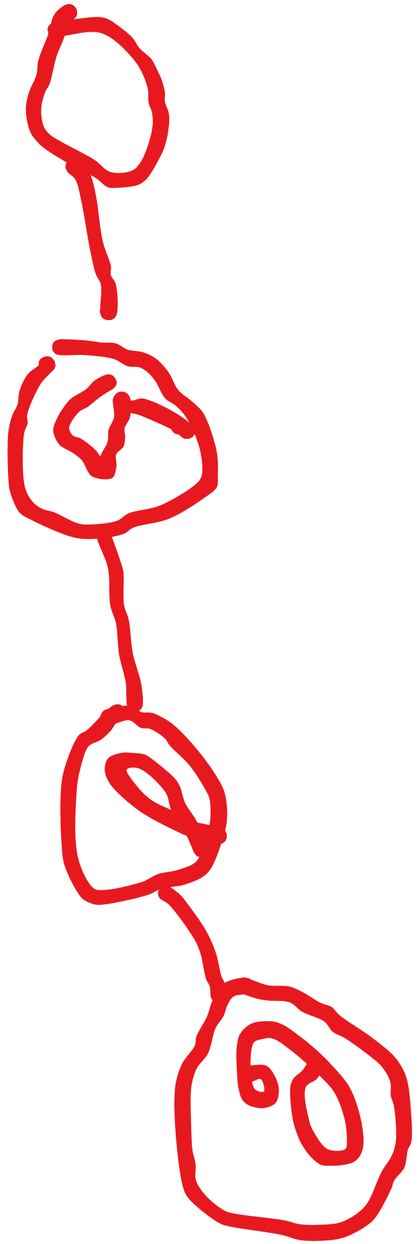


algorithom

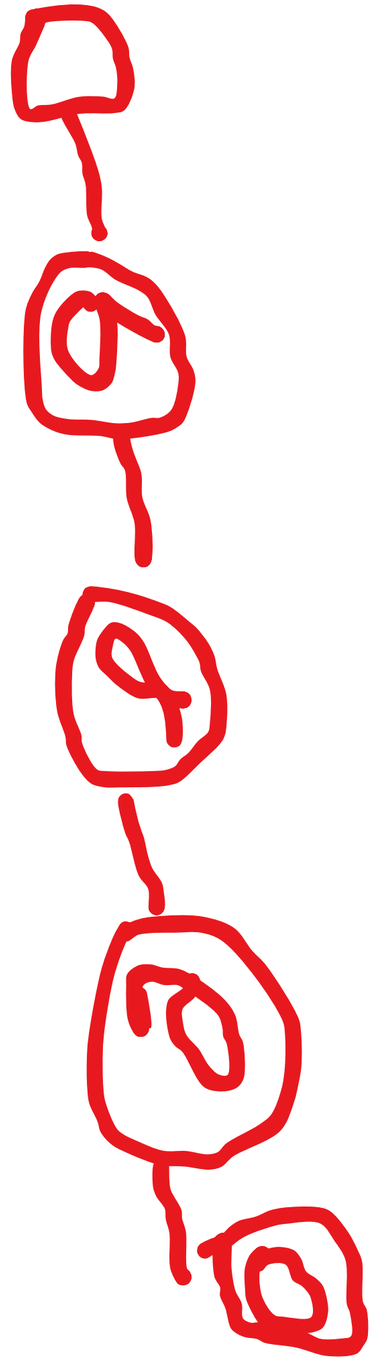
algorithom



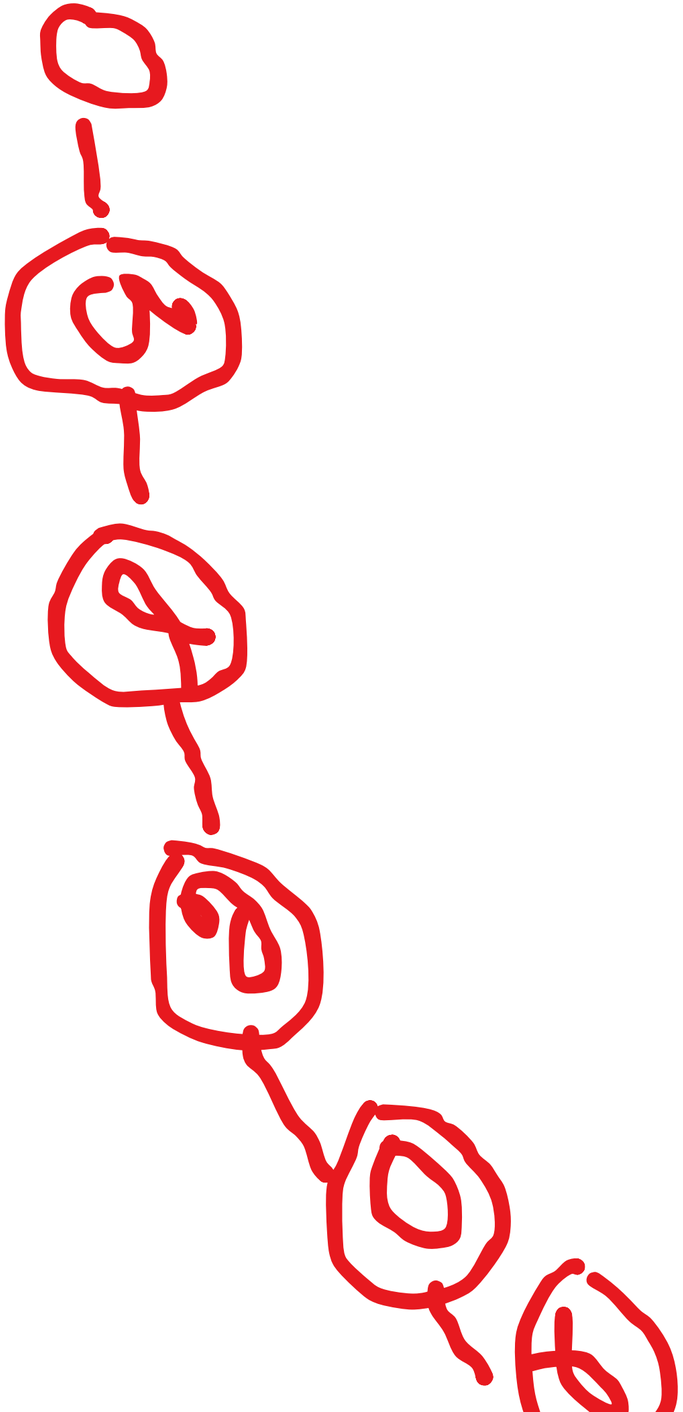
algorithom



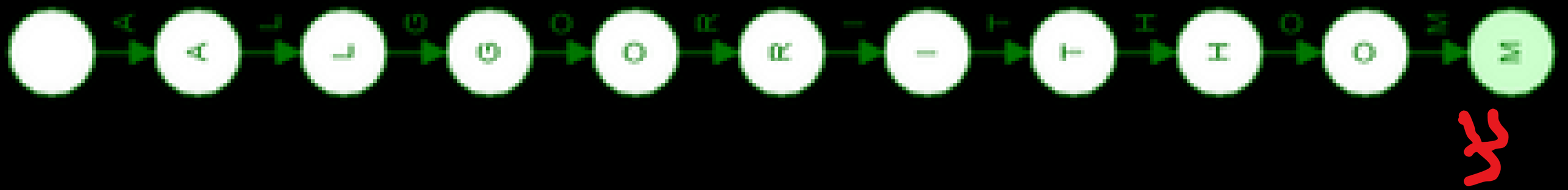
algorithom



algorithom

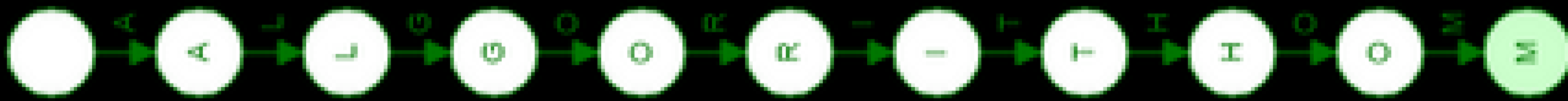


Algorithom



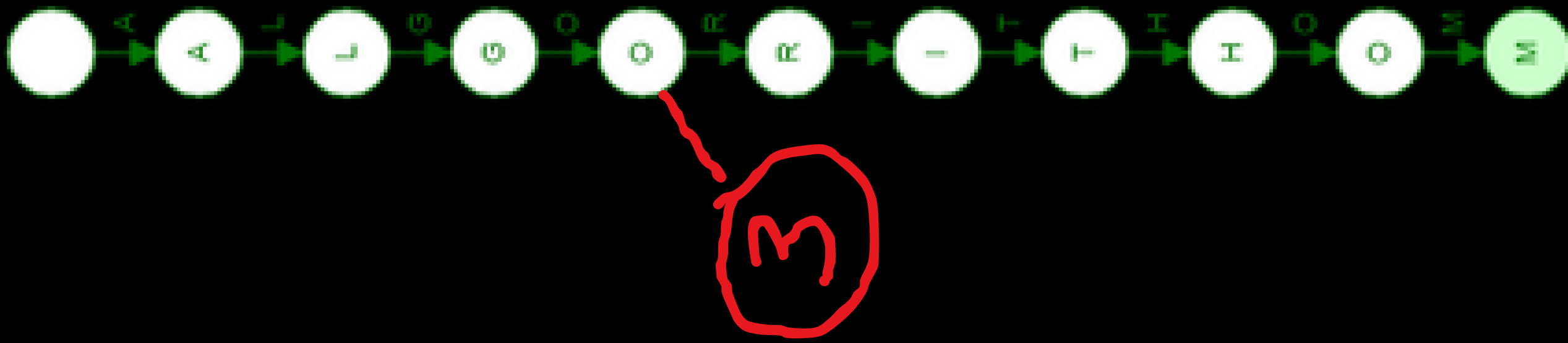
ᐊᓐᓀᐦᐃᑎᐦᐅᒪ

Algorithom



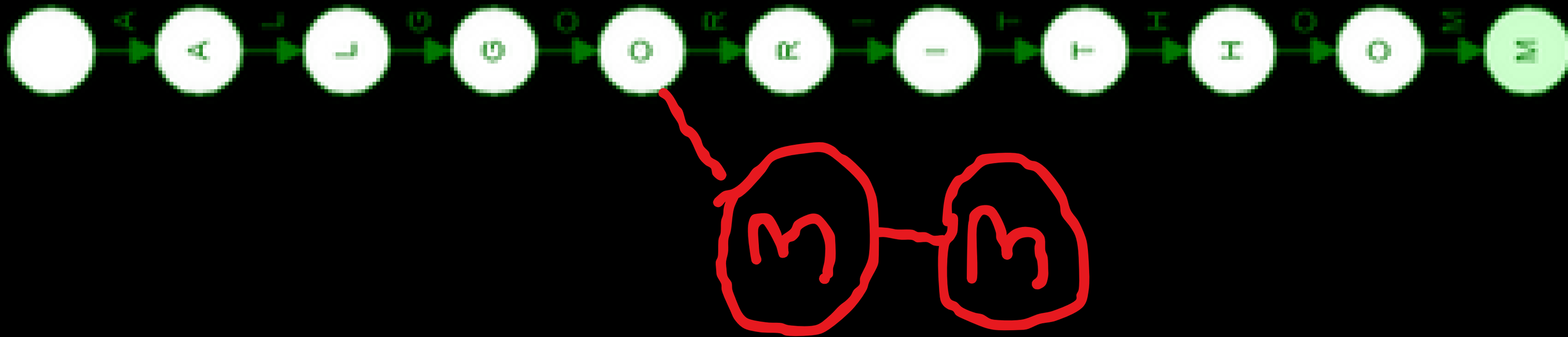
Algorithom

Algorithm

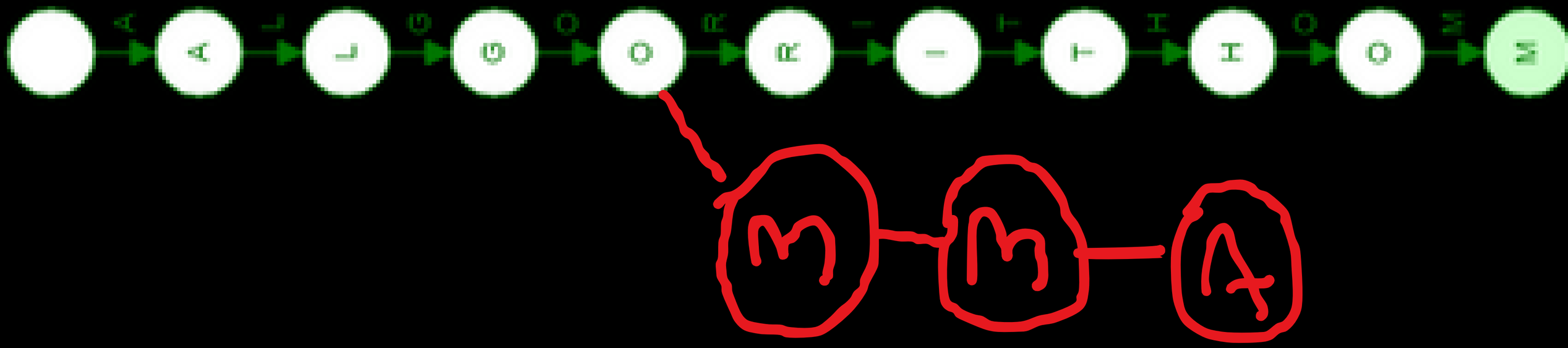


ᐊᓐᒃᓗᓕᖅᑐᑎᒃ

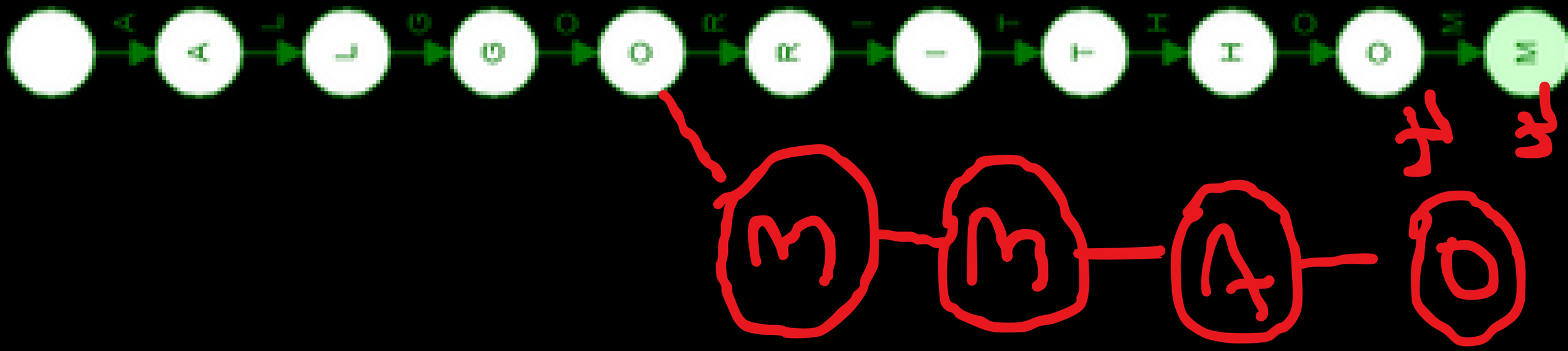
Algorithom



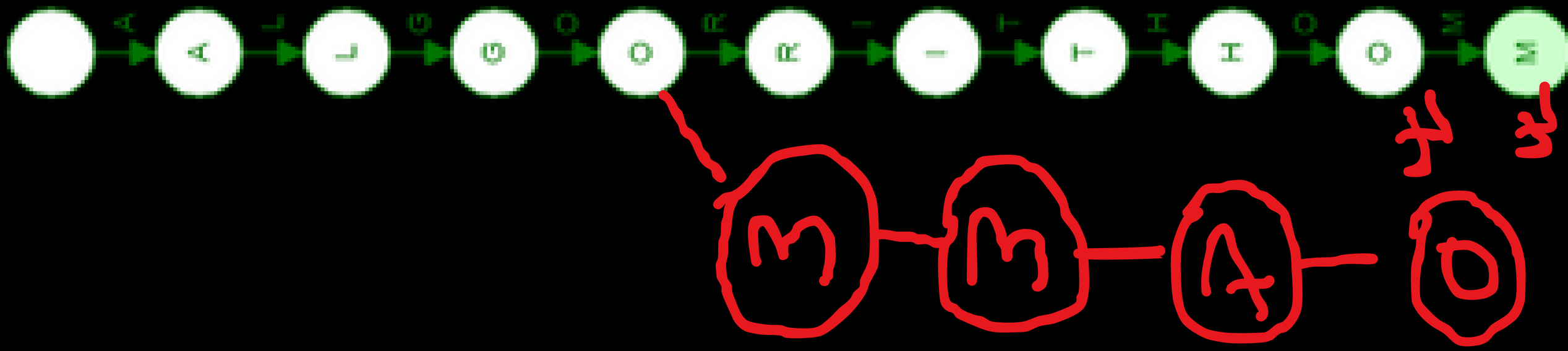
Algorithm



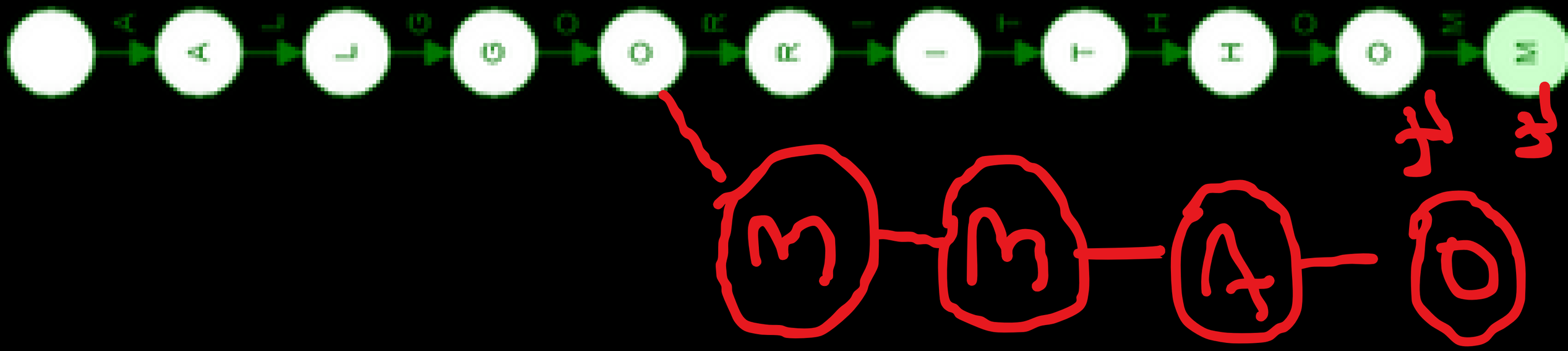
Algorithm



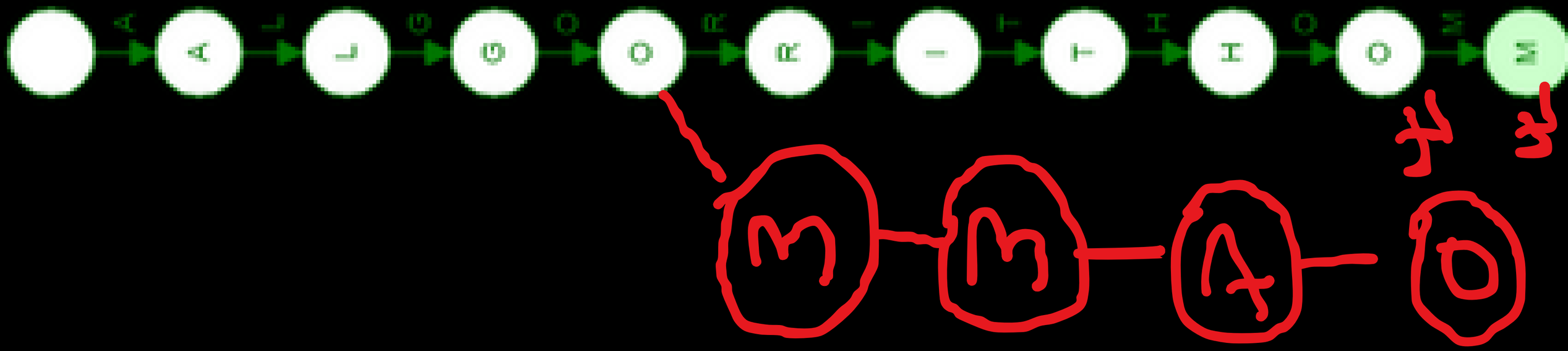
Algorithm



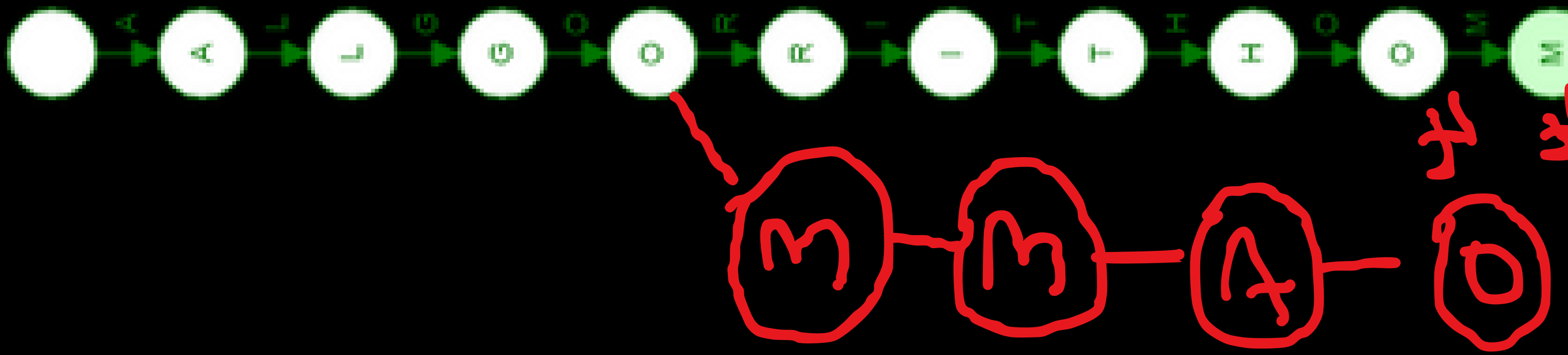
Algorithm



Algorithm

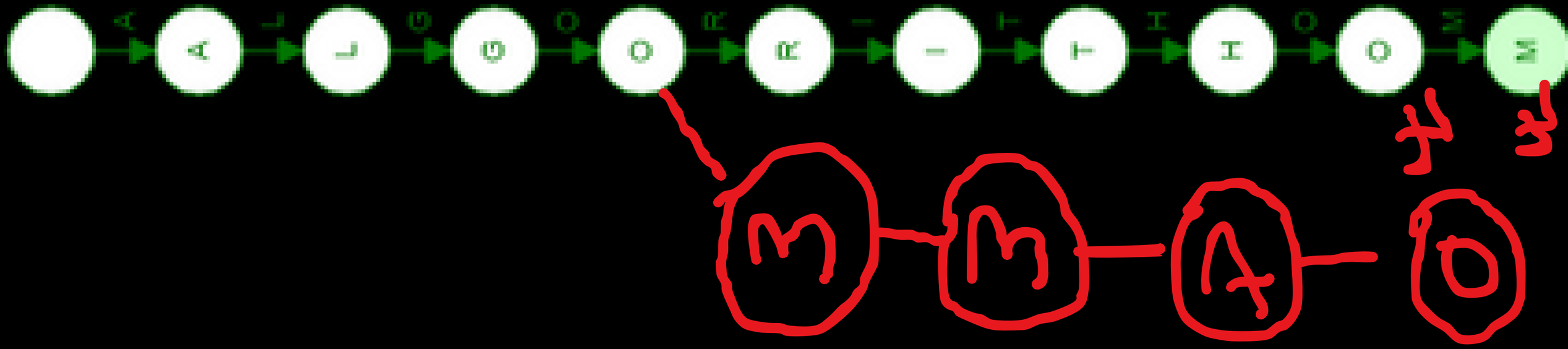


Algorithm

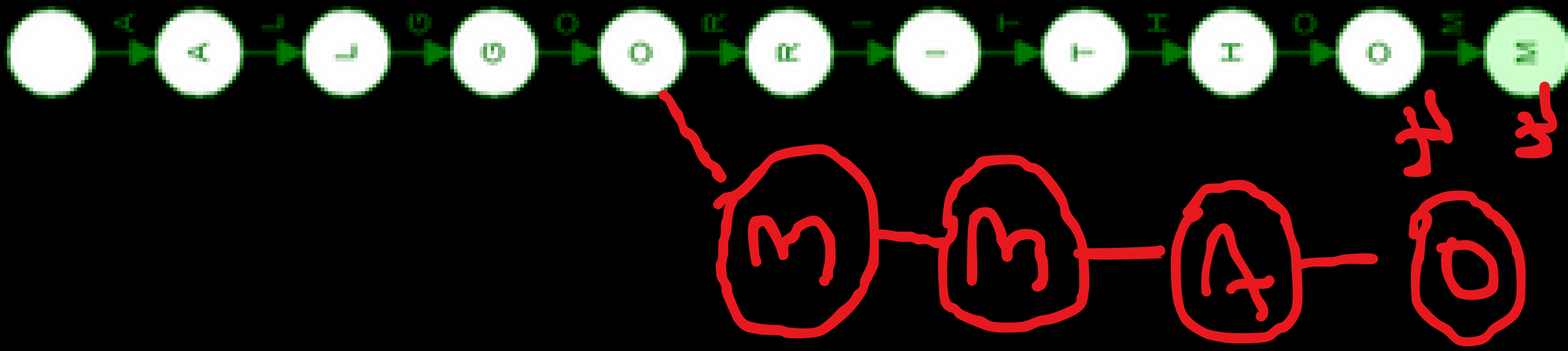


Algorithm

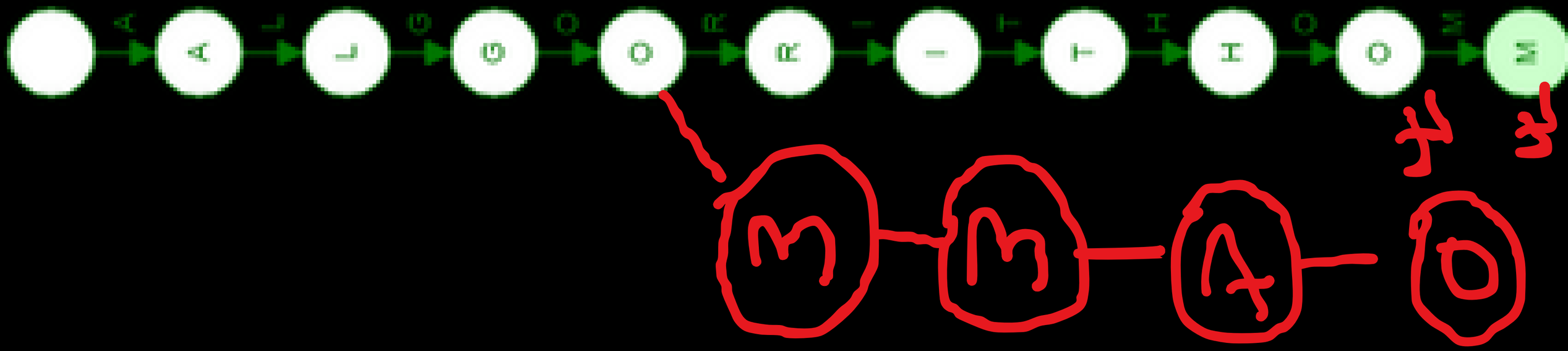
Algorithm



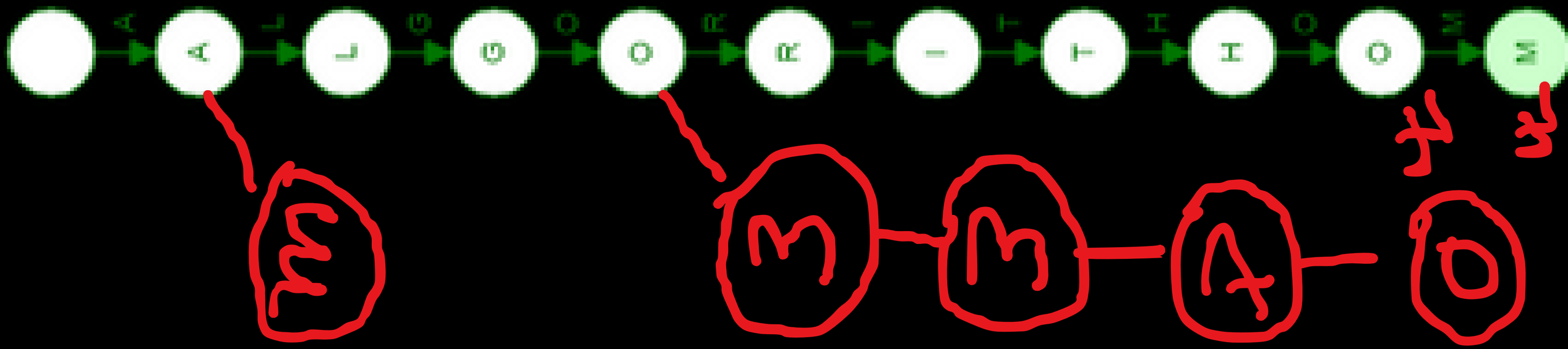
Algorithm



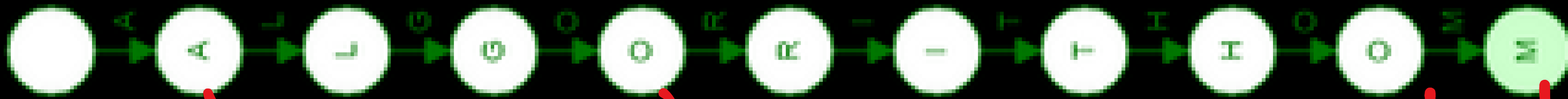
Algorithm



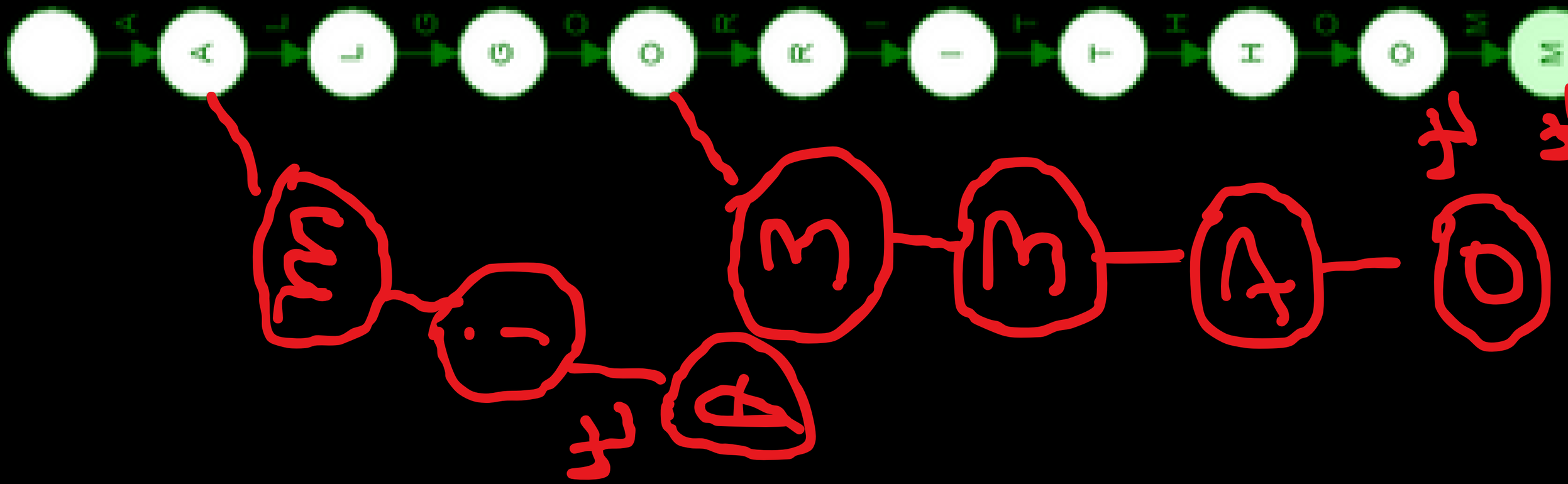
Algorithm



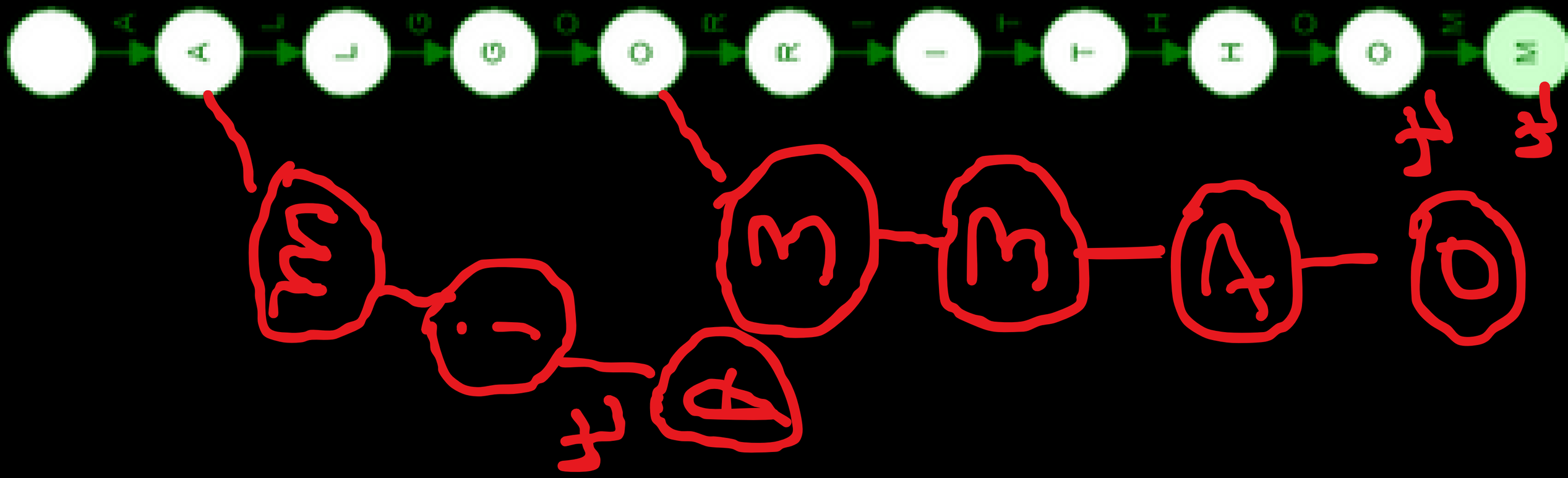
Algorithm



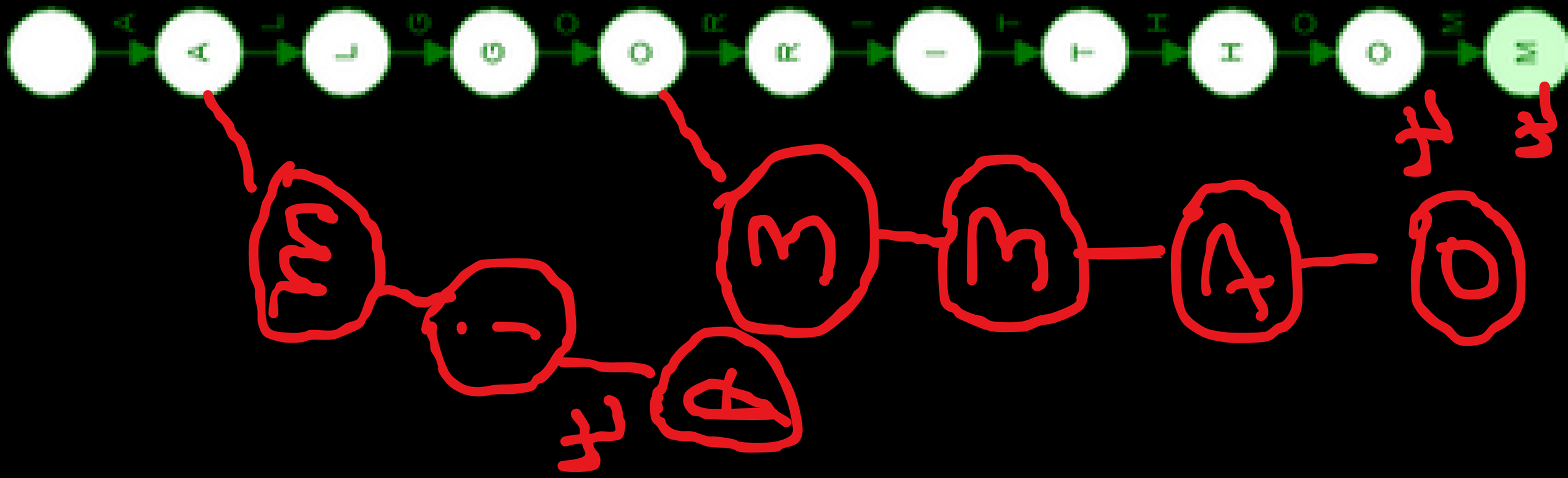
Algorithm



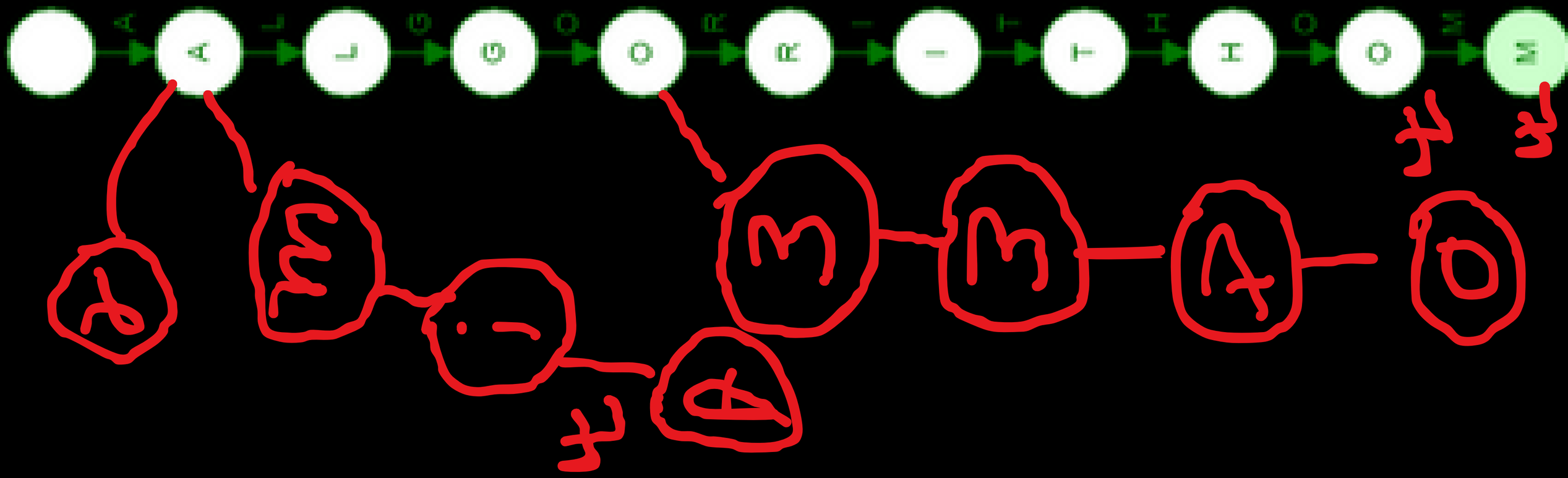
Algorithm



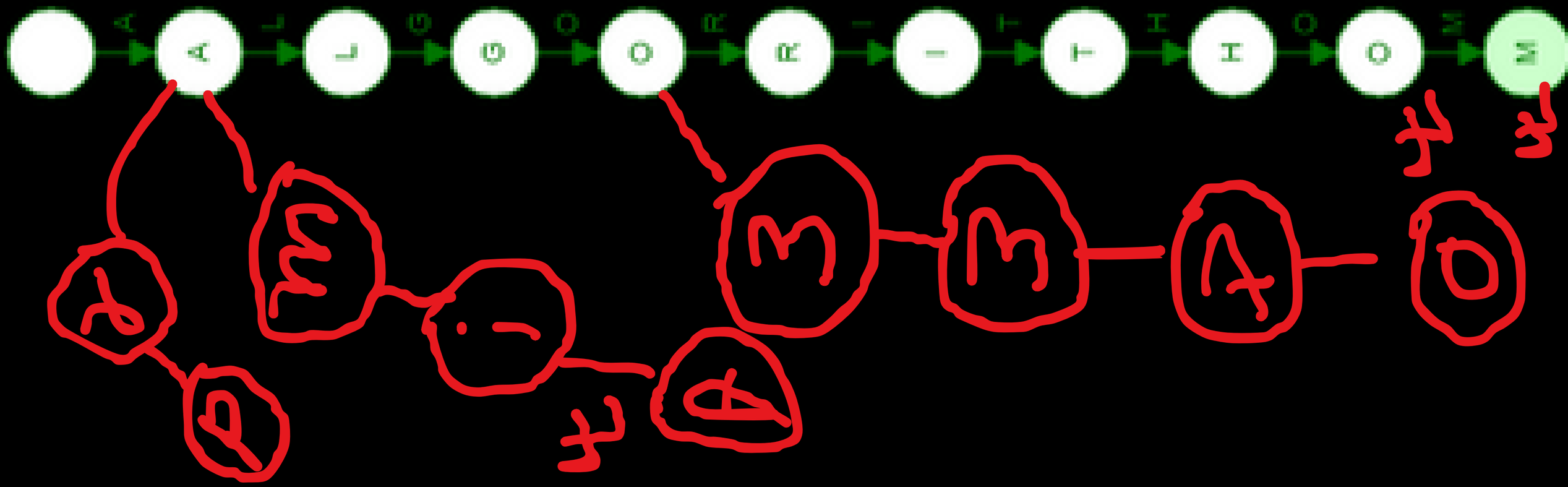
Algorithm



Algorithm



Algorithm

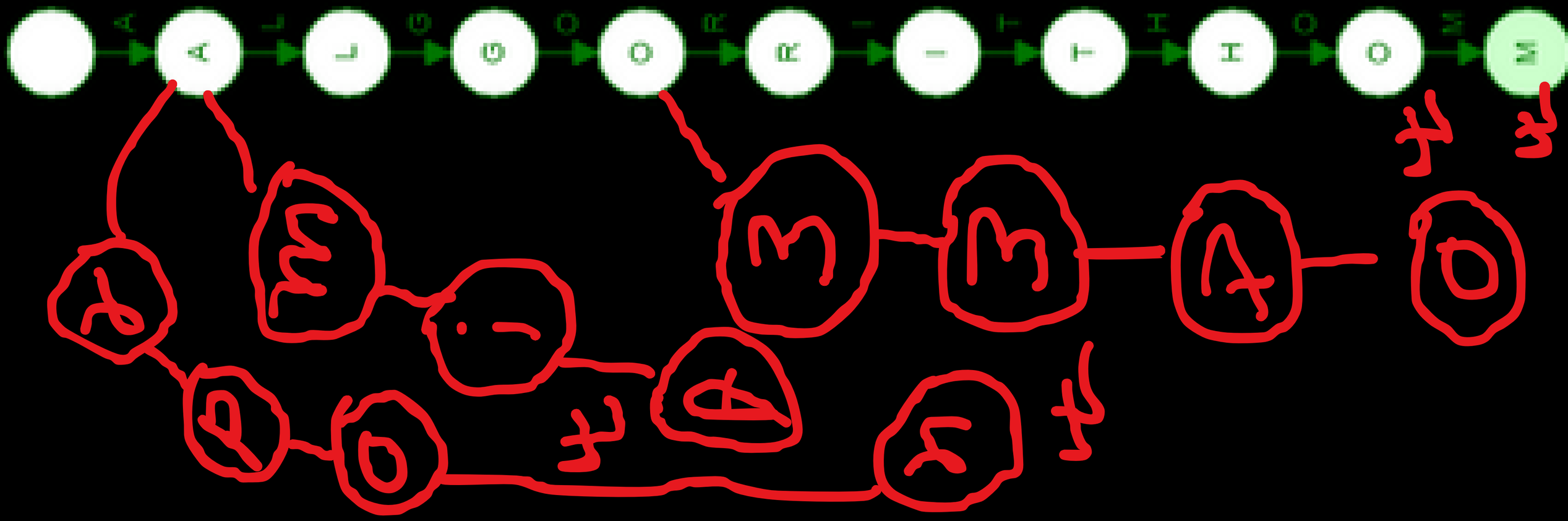


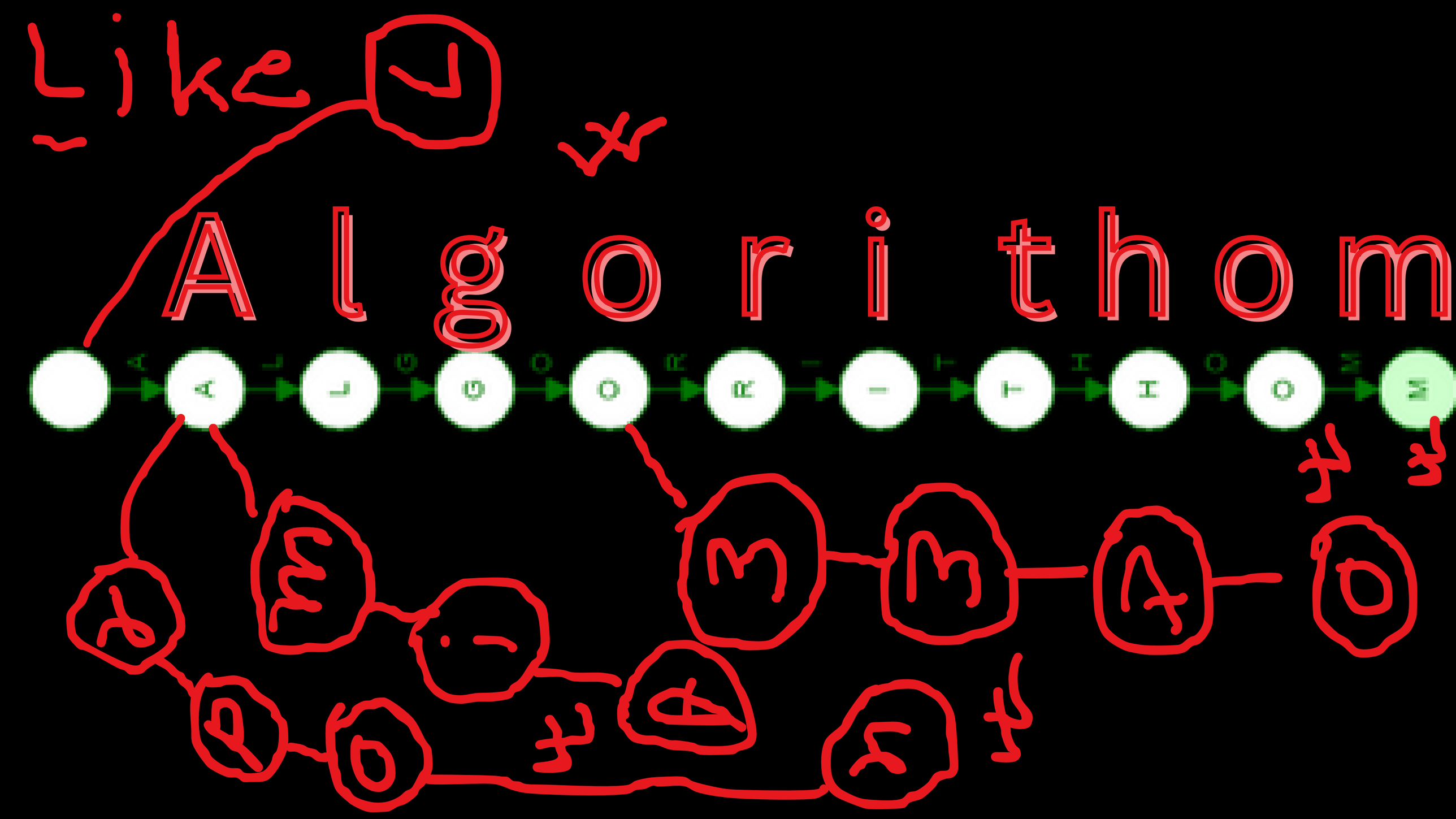
Algorithm

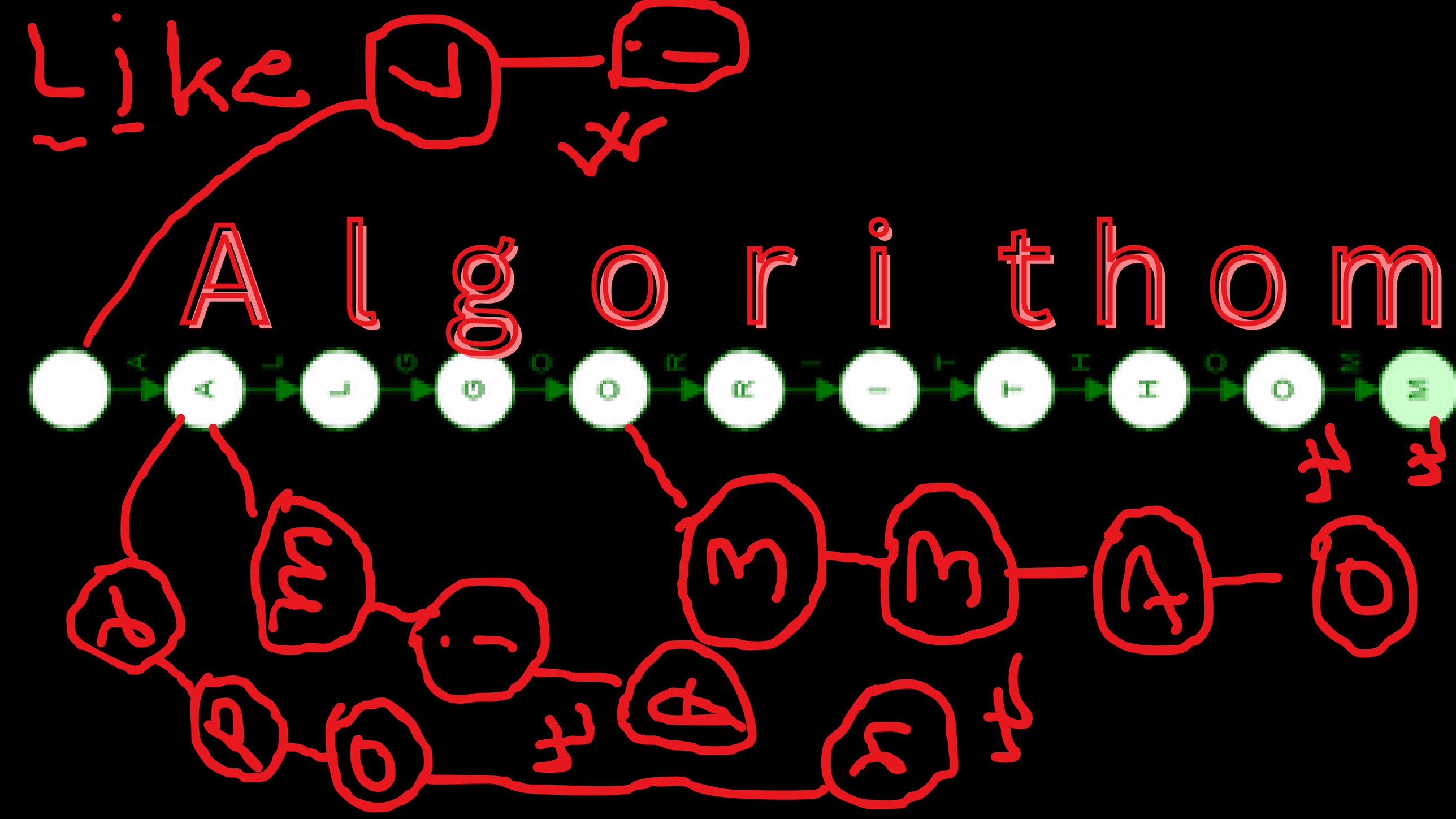
Algorithom

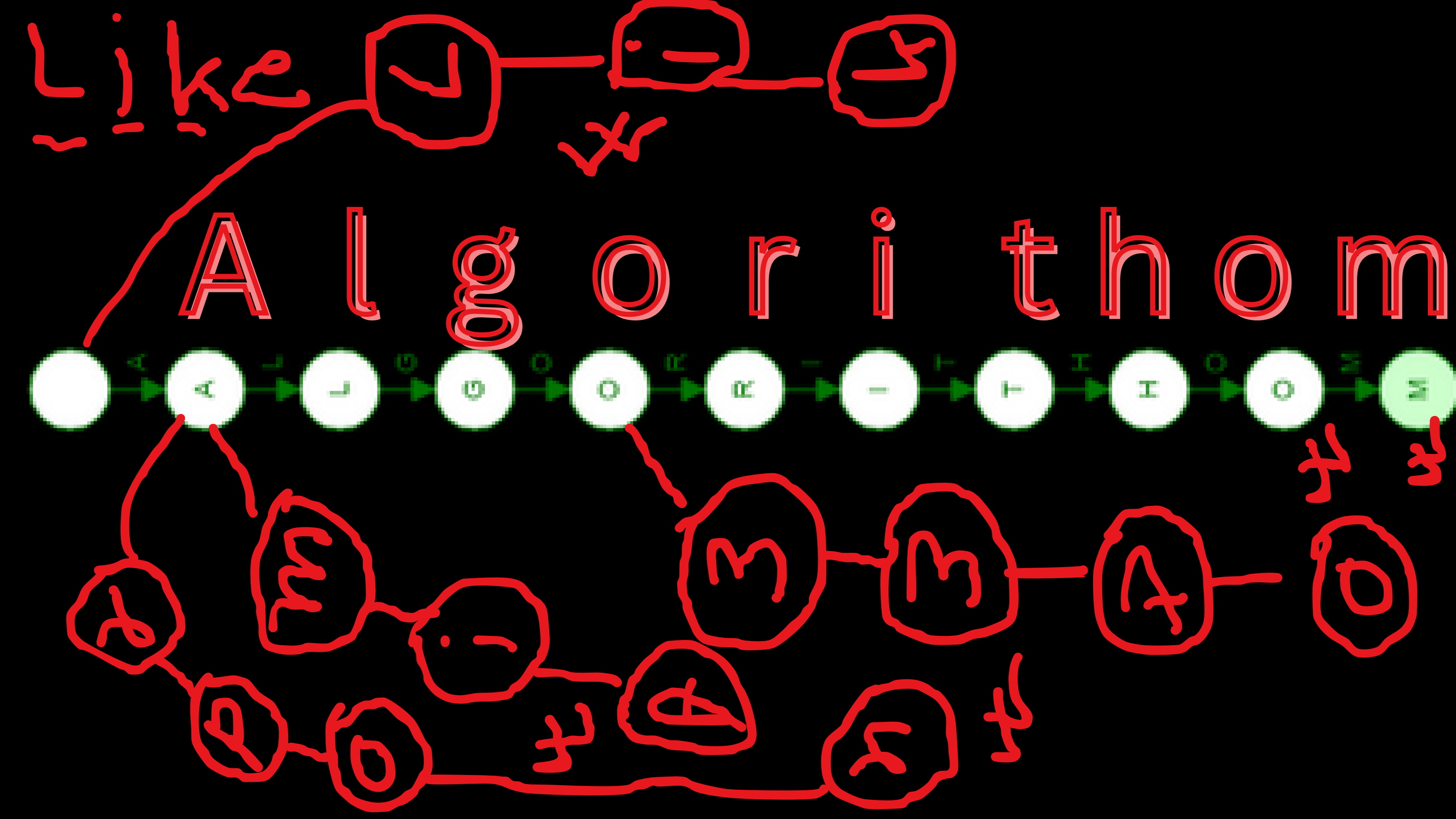
Like

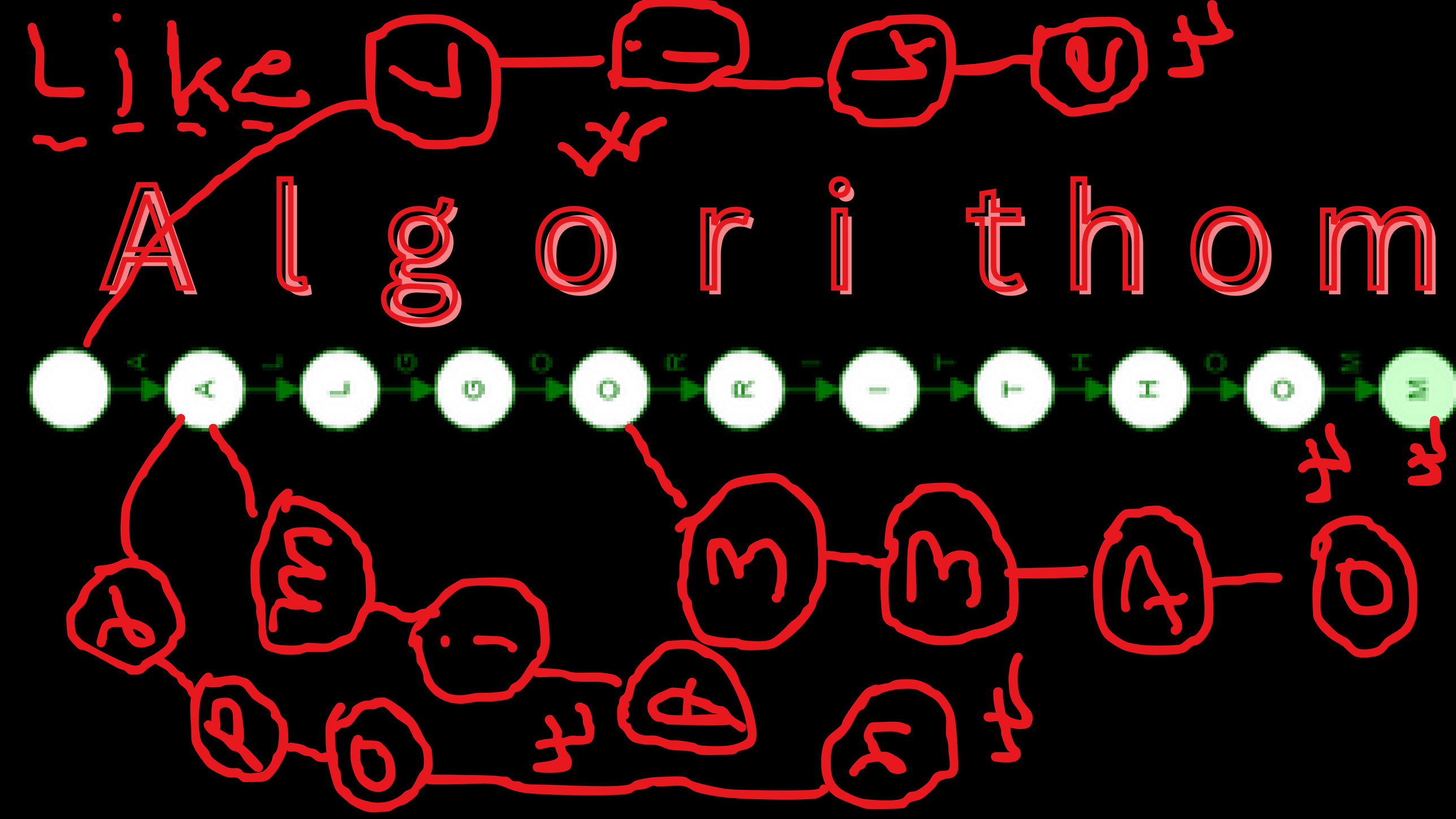
Algorithom




Like
Algorithom

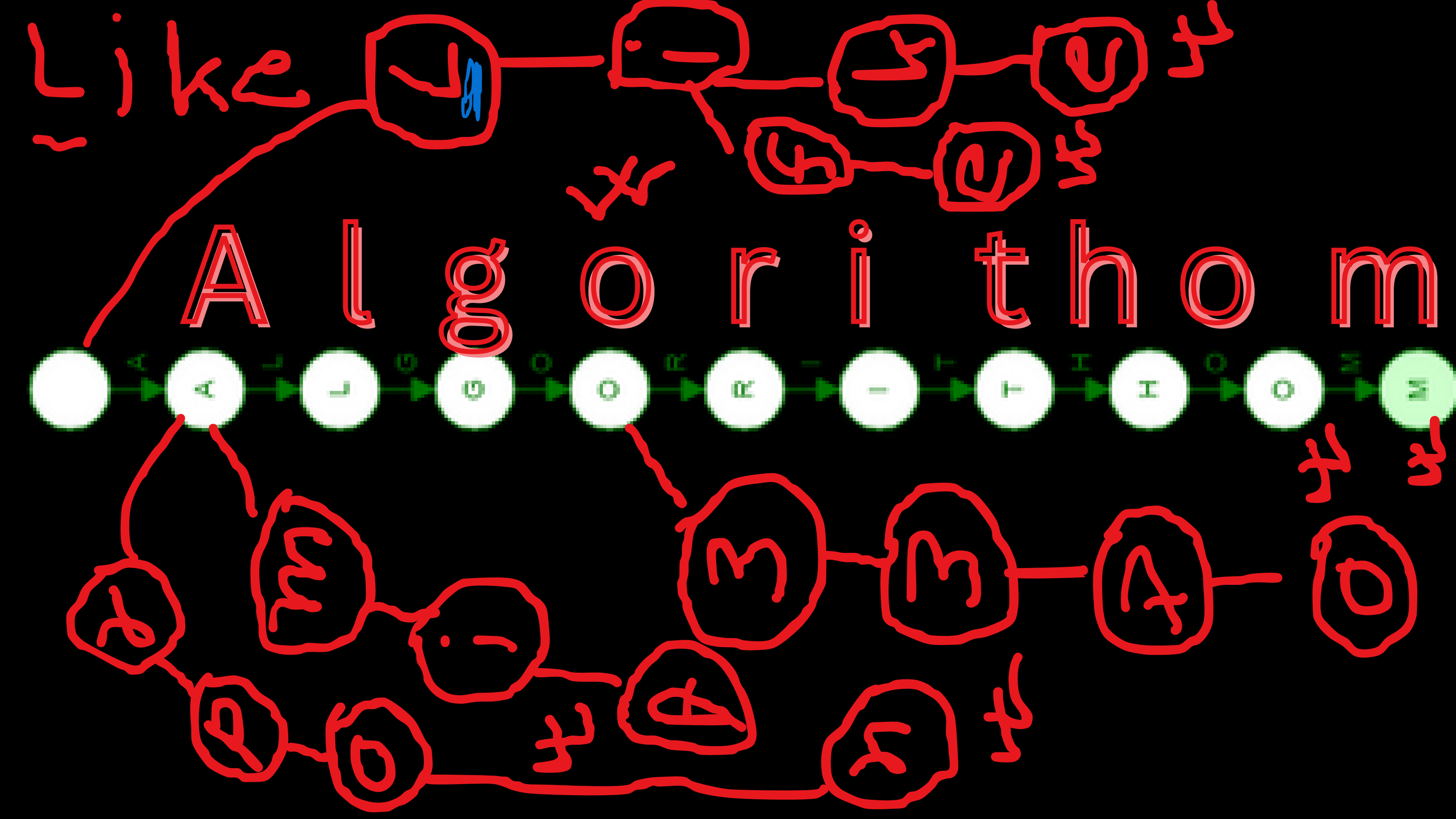

Like
Algorithom

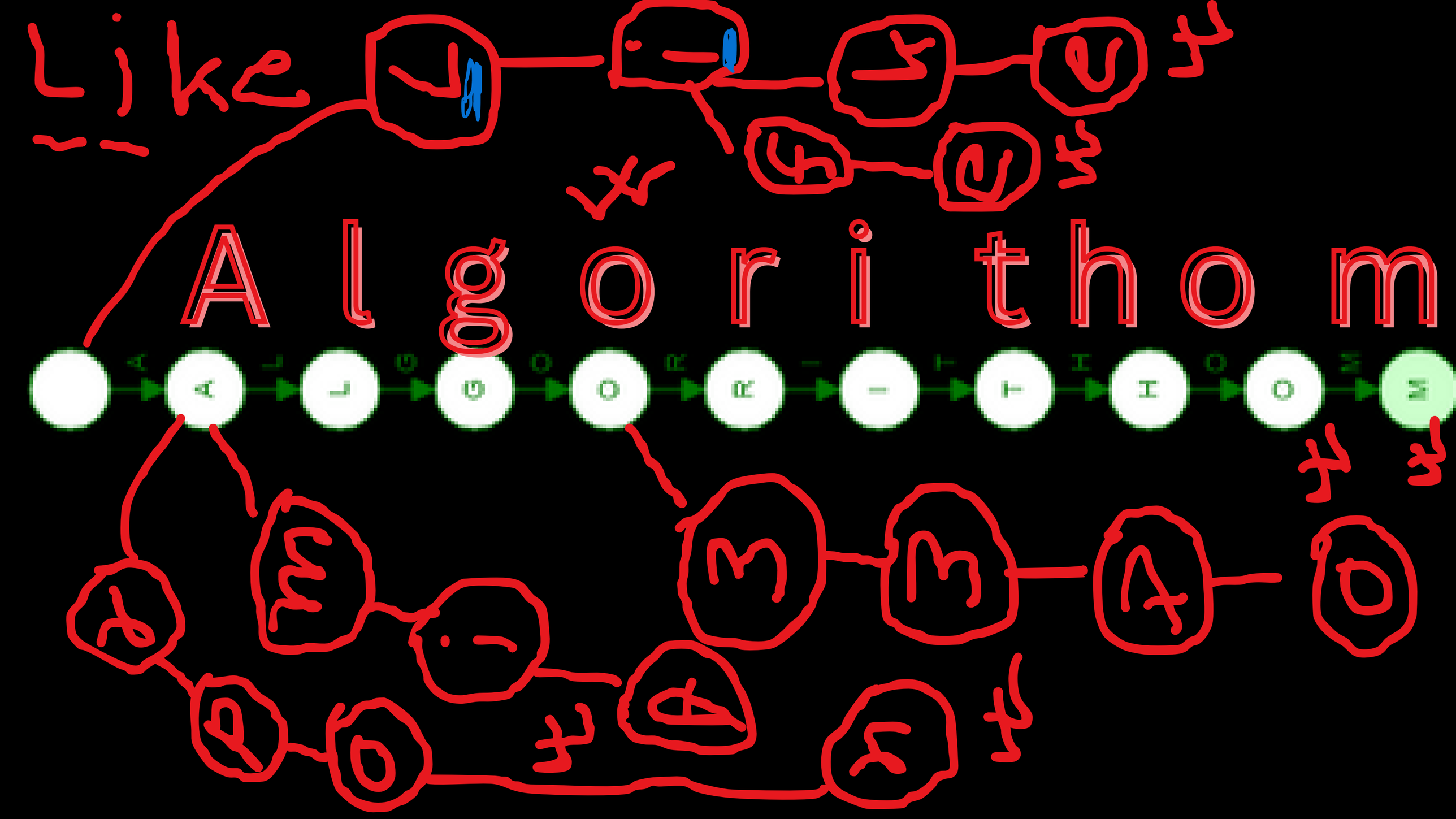

Like
Algorithom


Like
Algorithom

Lets try to add Life at here.

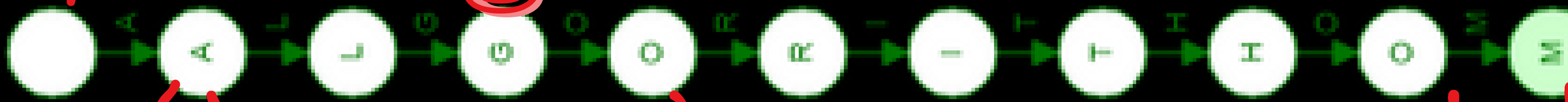
Now, try to search from here.

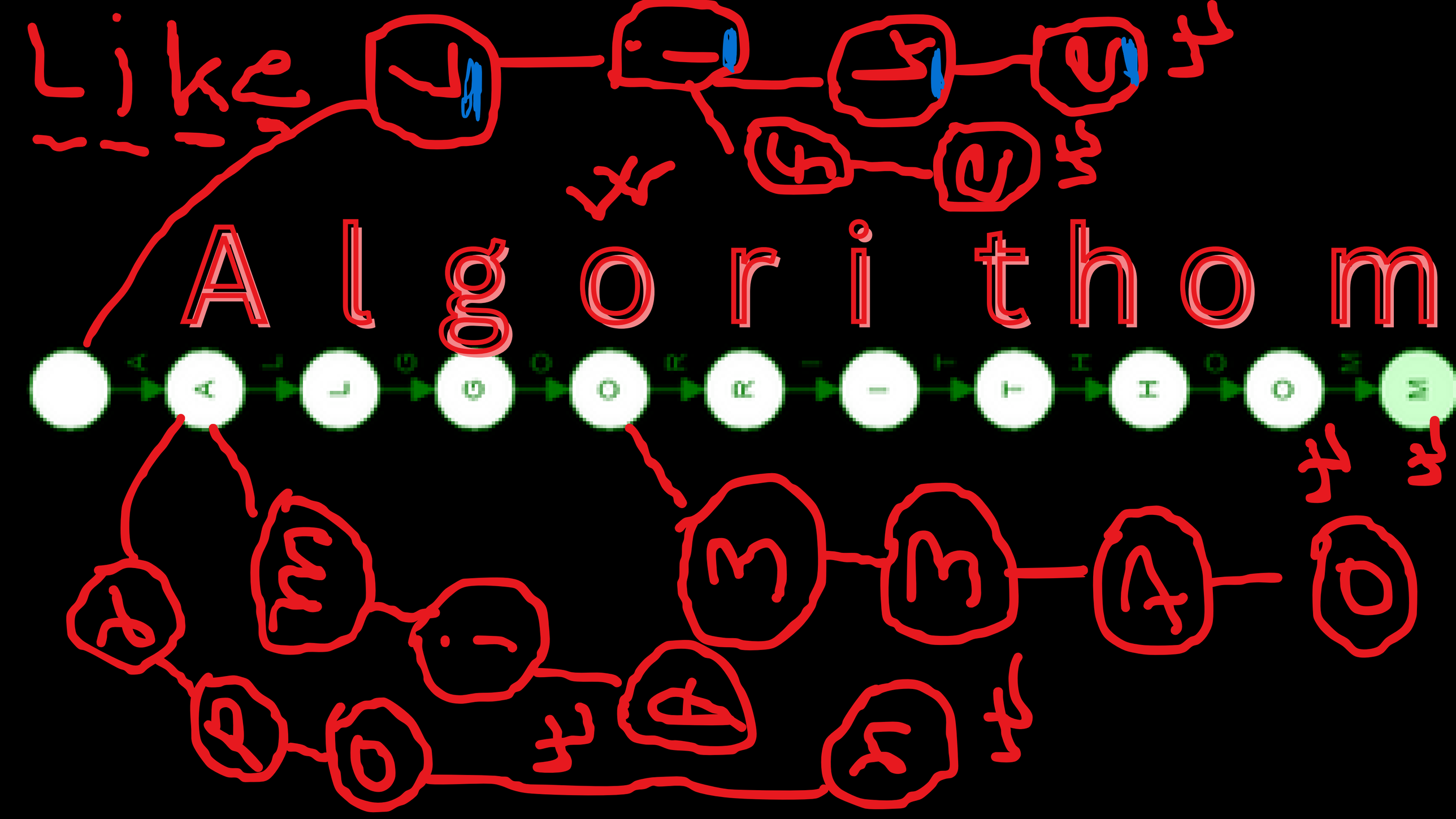

Like
Algorithom


Like
Algorithom

Like

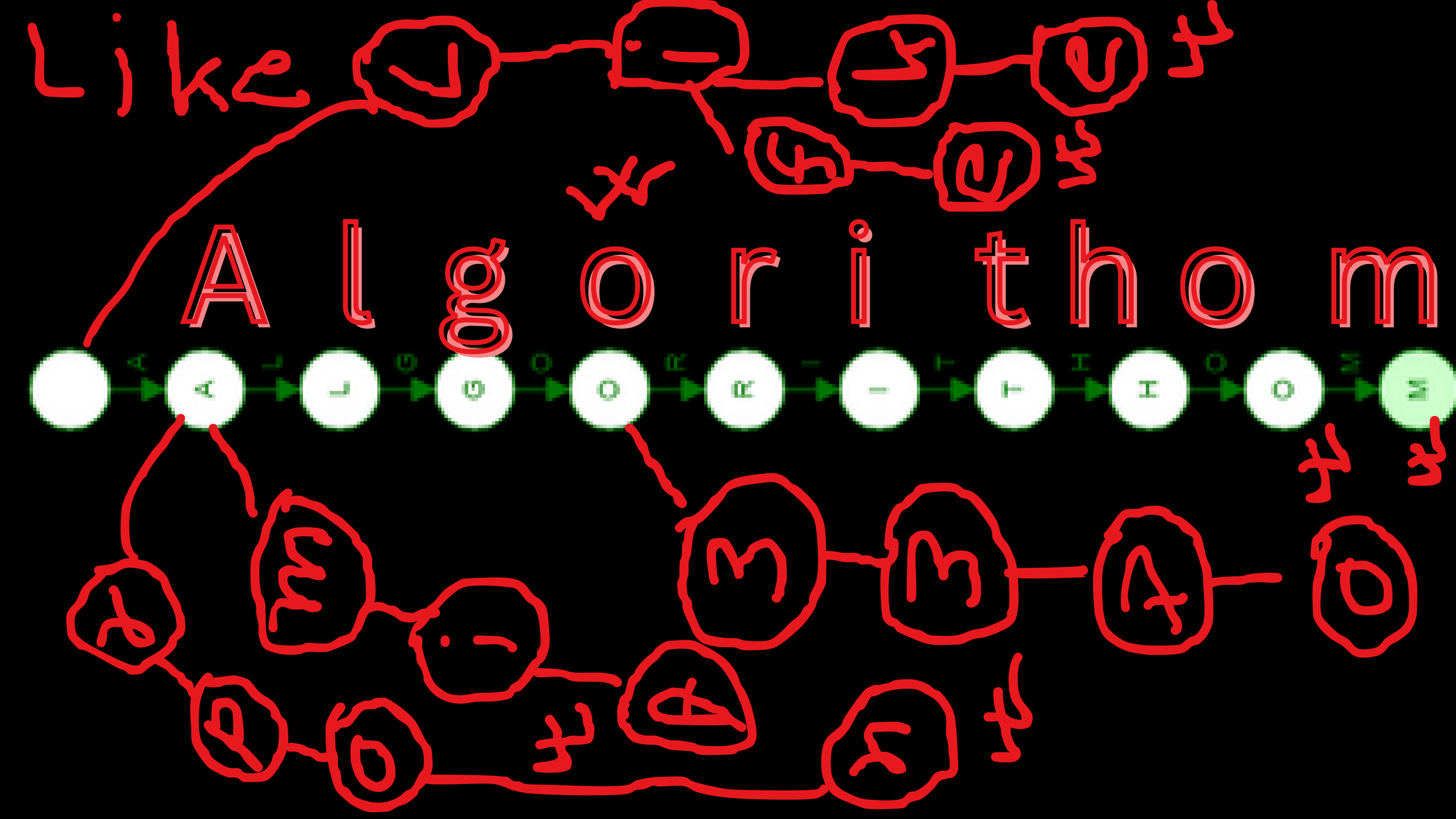
Algorithom

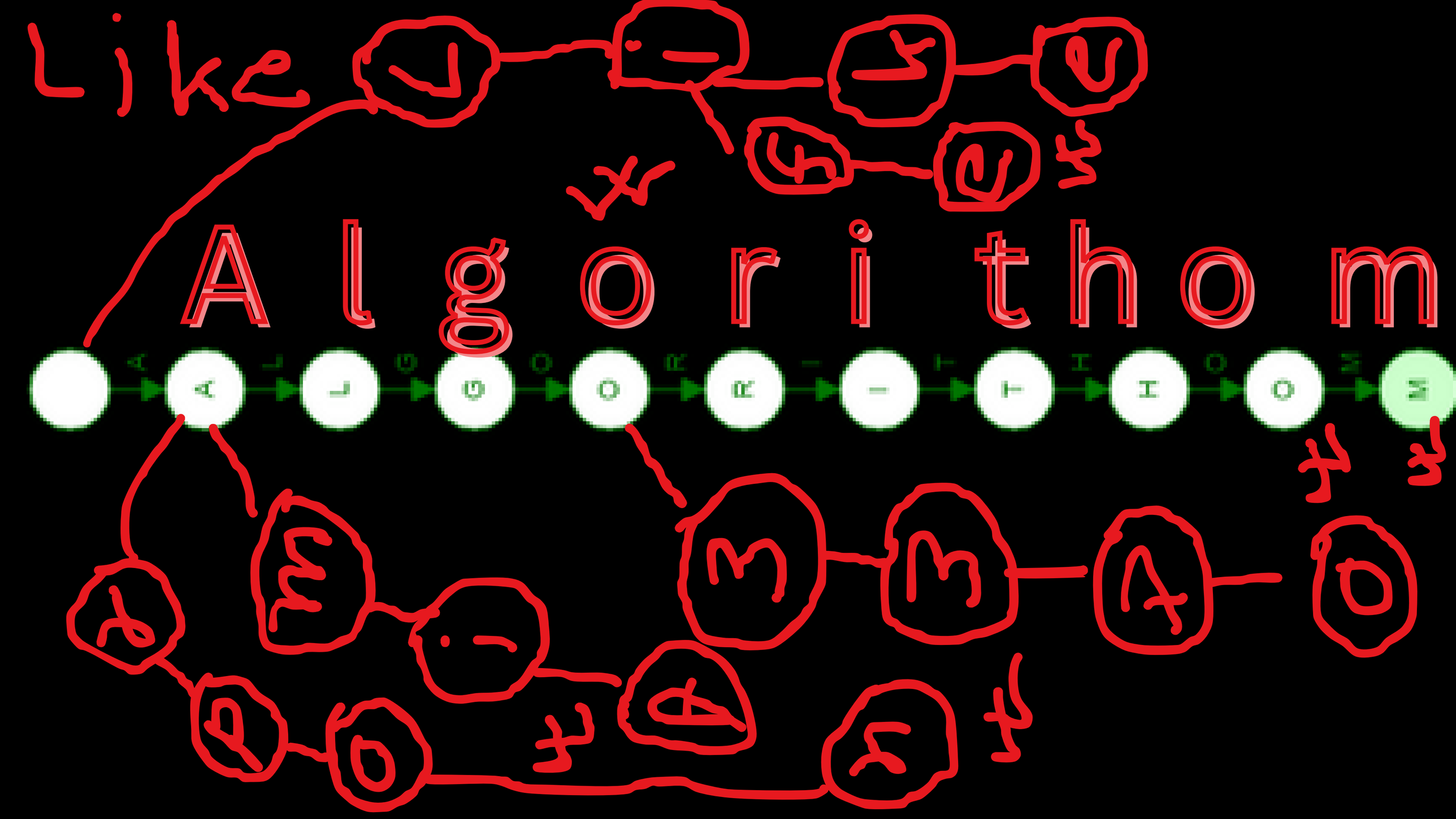


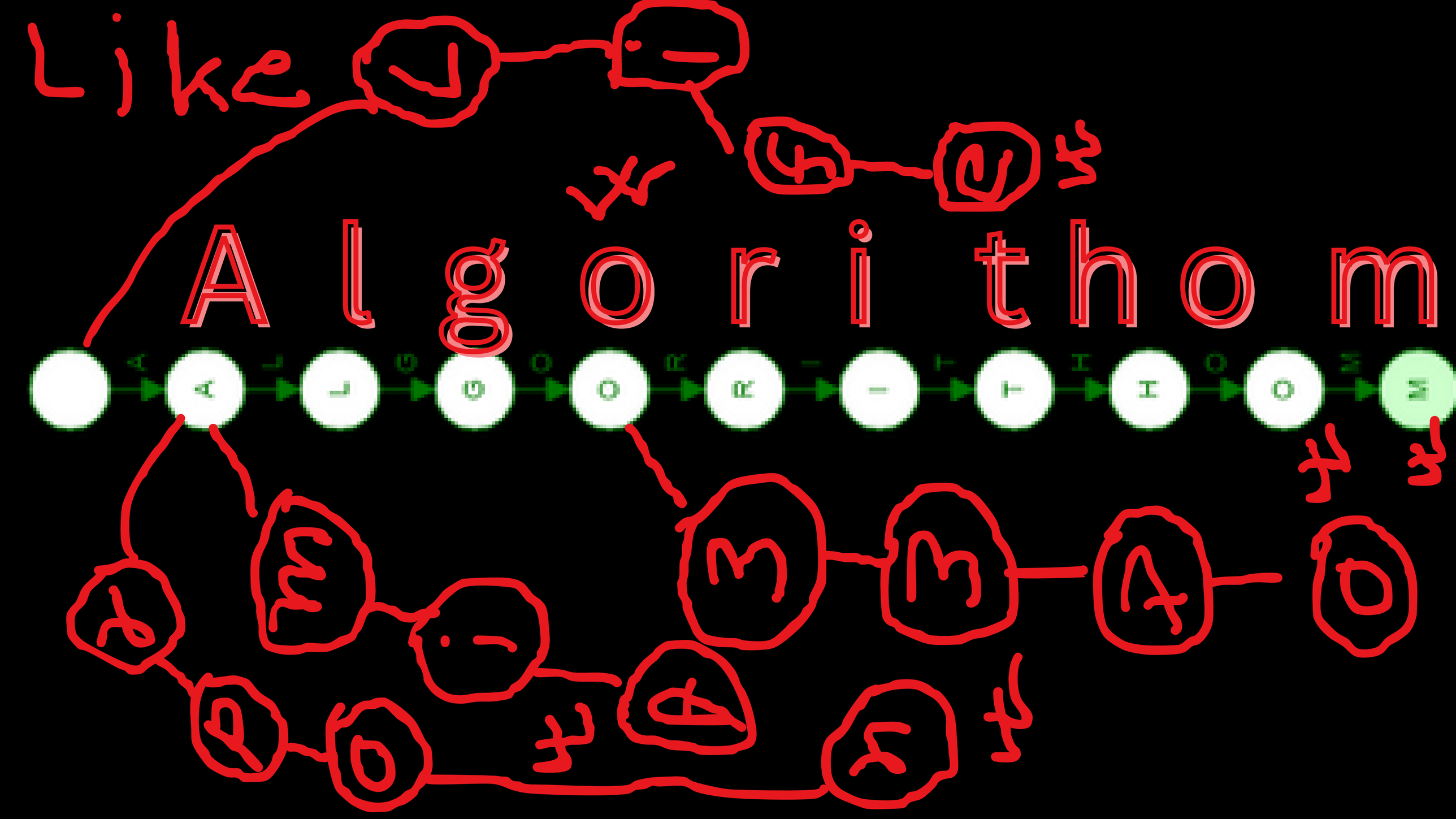

Like
Algorithom

Lets try to delete some variable from here.

In case of deletion we mainly not delete a string. we mainly delete end character of the string as false. And we mainly delete the char’s which is not present as a prefix of any string.


Like
Algorithom


Like
Algorithom


Like
Algorithom

Let’s go to the implementations.