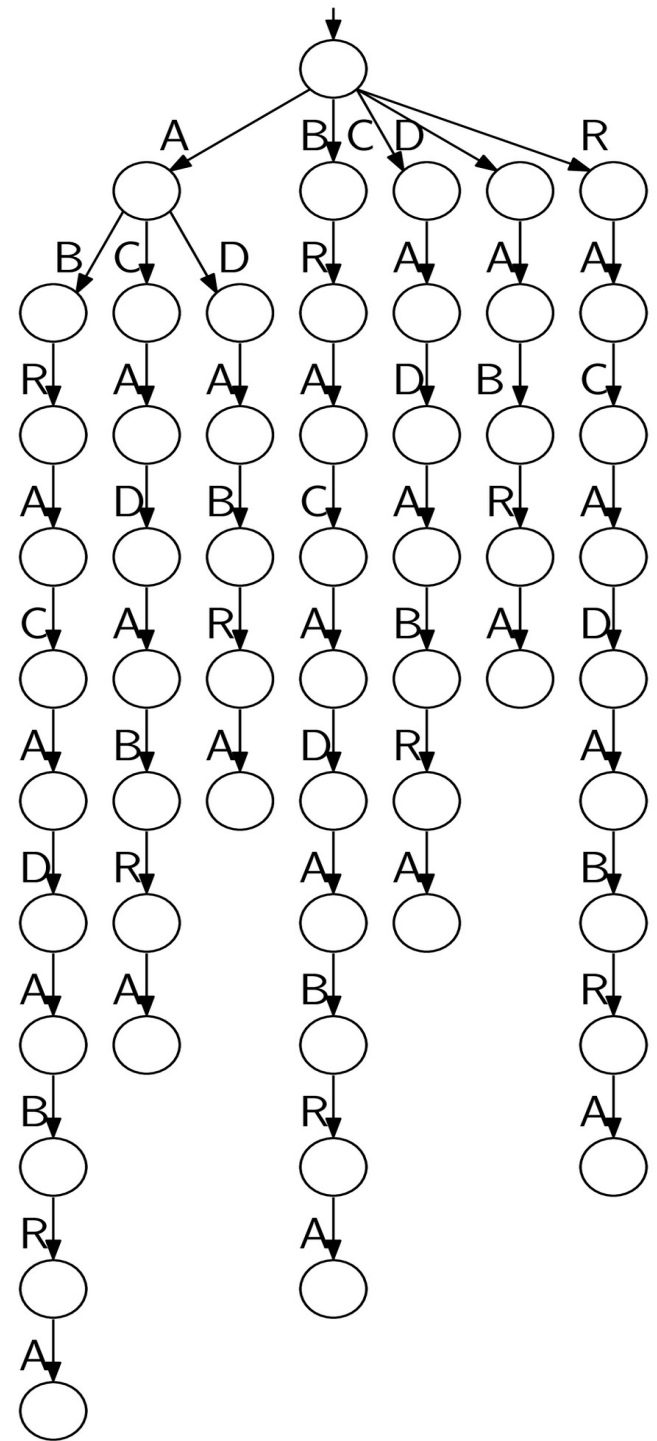# Introduction to Wheeler Graphs

Giovanni Manzini

Searching for substrings of ABRACADABRA

We can use a trie
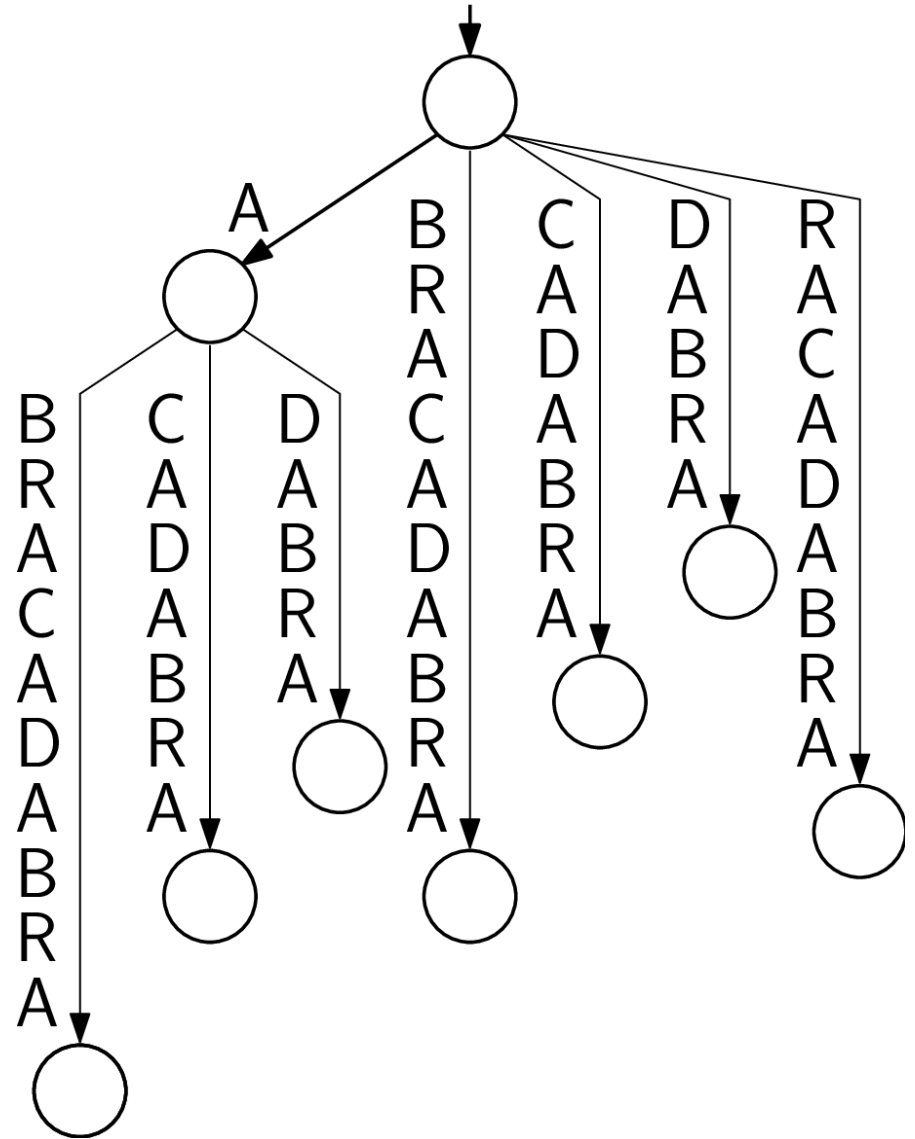
Simple but not space efficient

Searching for substrings of ABRACADABRA
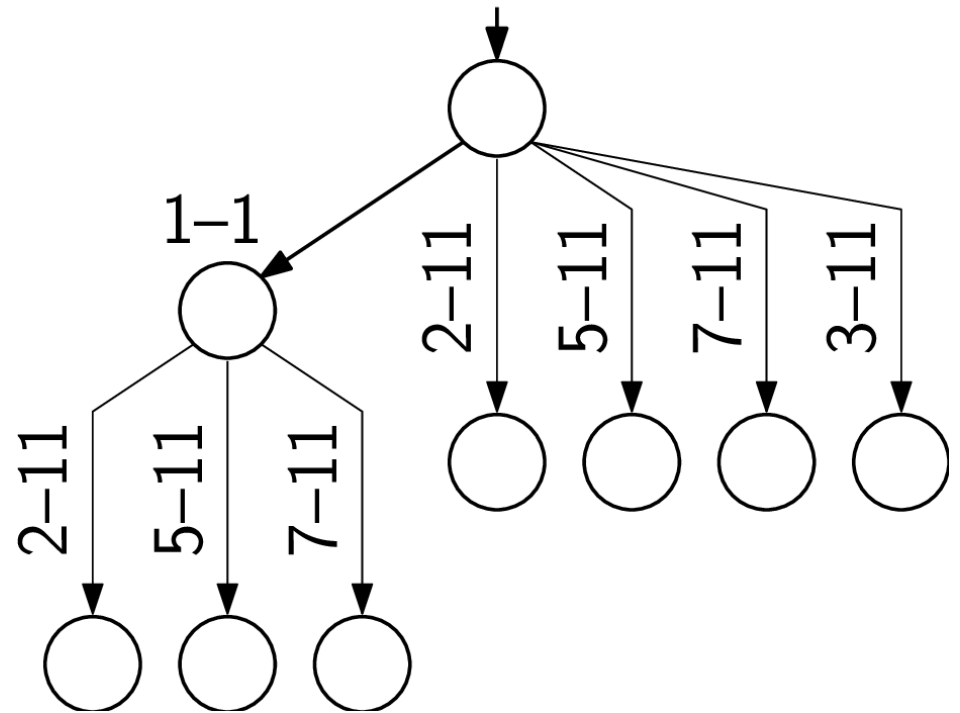
Compacted Trie

More space efficient!

Searching for substrings of ABRACADABRA

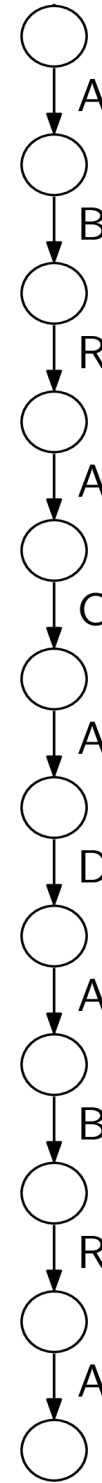

Suffix Tree

"theoretically" space efficient

Searching for substrings of ABRACADABRA

We can use a graph
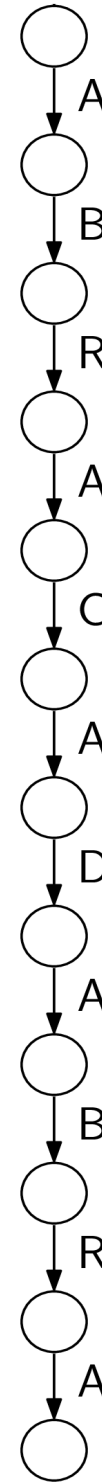and search for subpaths

Extremely space efficient!

Searching for substrings of ABRACADABRA

We can use a graph
and search for subpaths
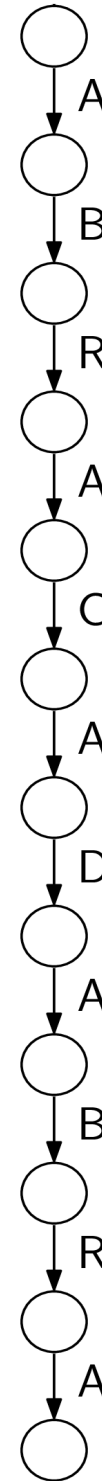
Extremely space efficient!

Searching is a headache

A
B
R
A
C
A
D
A
B
R
A

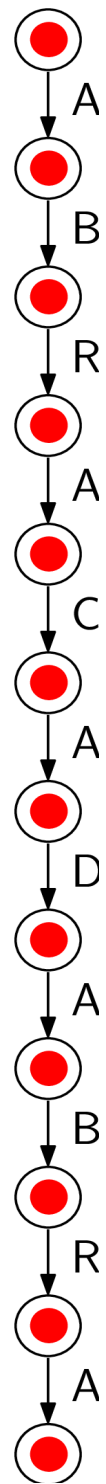Searching for substrings
of ABRACADABRA

Example:
Searching ABR

Searching all paths labeled
ABR is equivalent to finding
the final nodes of the paths.
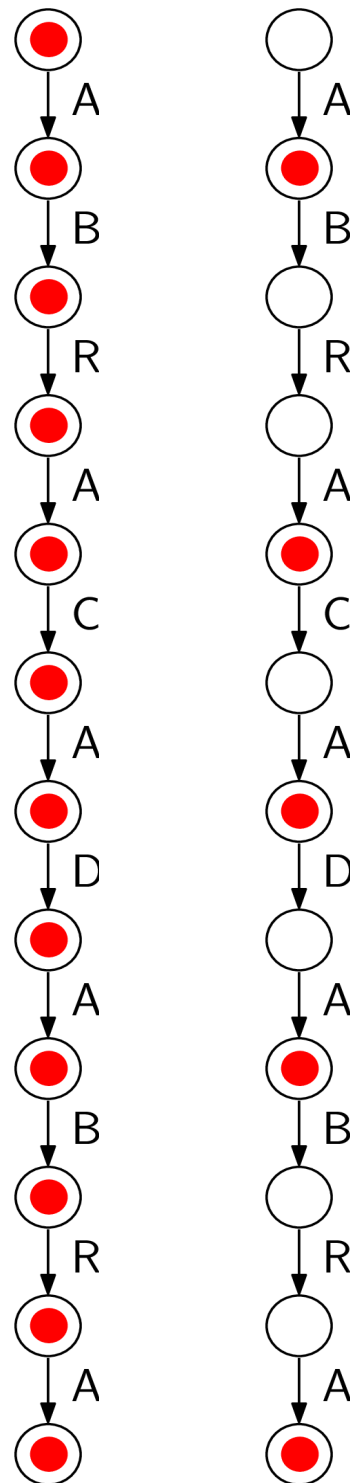
We proceed by searching
prefixes of size 0,1,2,3

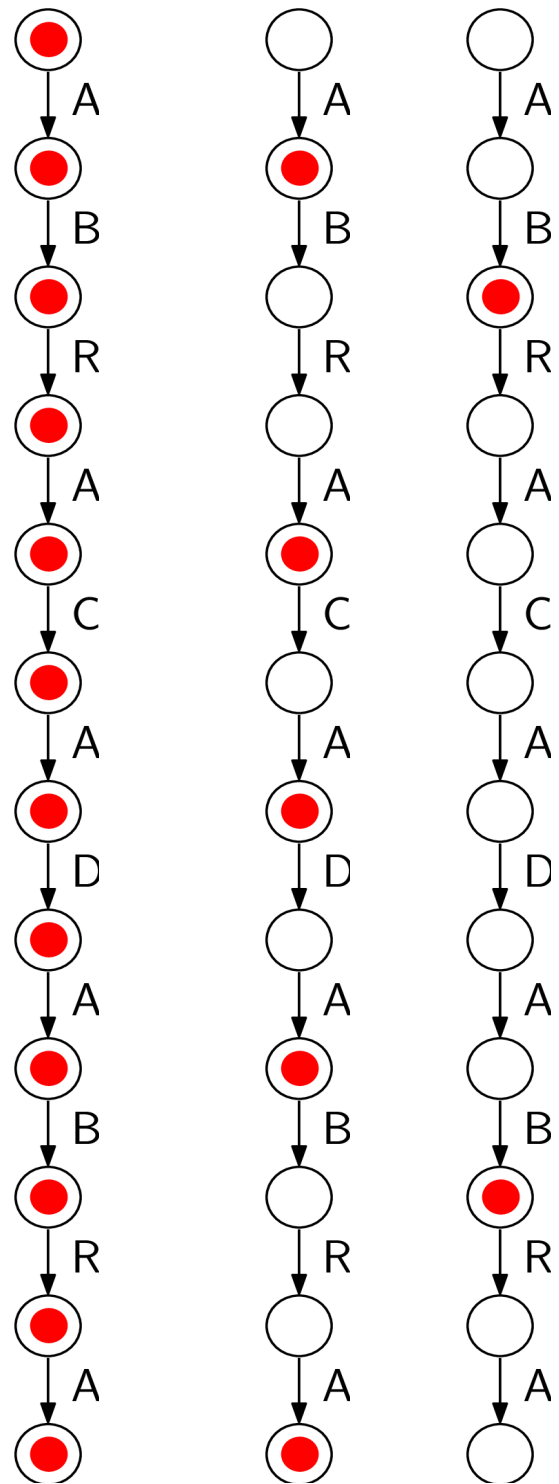Searching for substrings
of ABRACADABRA

Example:
Searching ABR

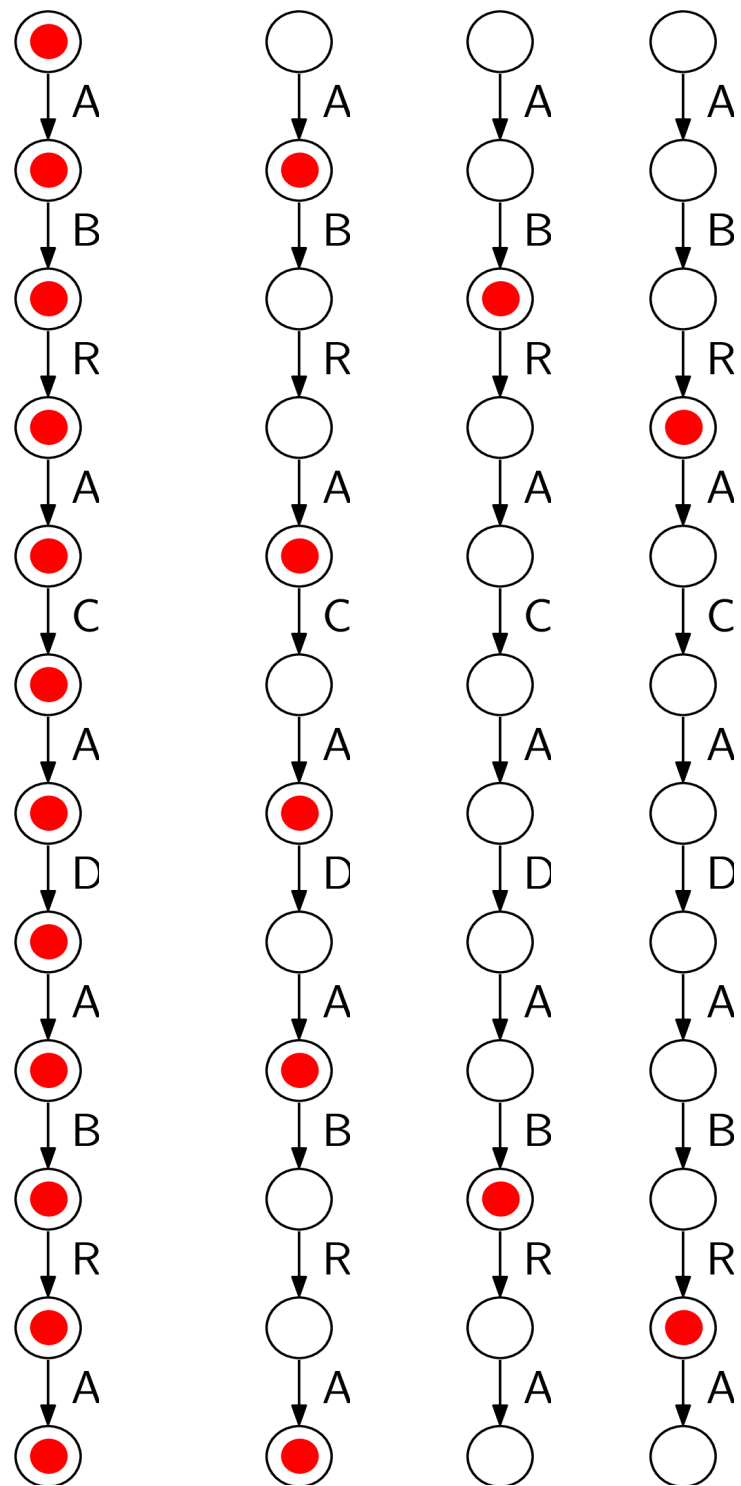Searching for substrings
of ABRACADABRA

Example:
Searching ABR

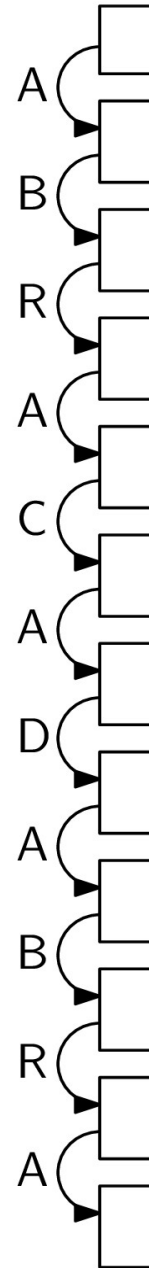# Searching for substrings of ABRACADABRA

Example:
Searching ABR

Searching for substrings of ABRACADABRA

Example:
Searching ABR

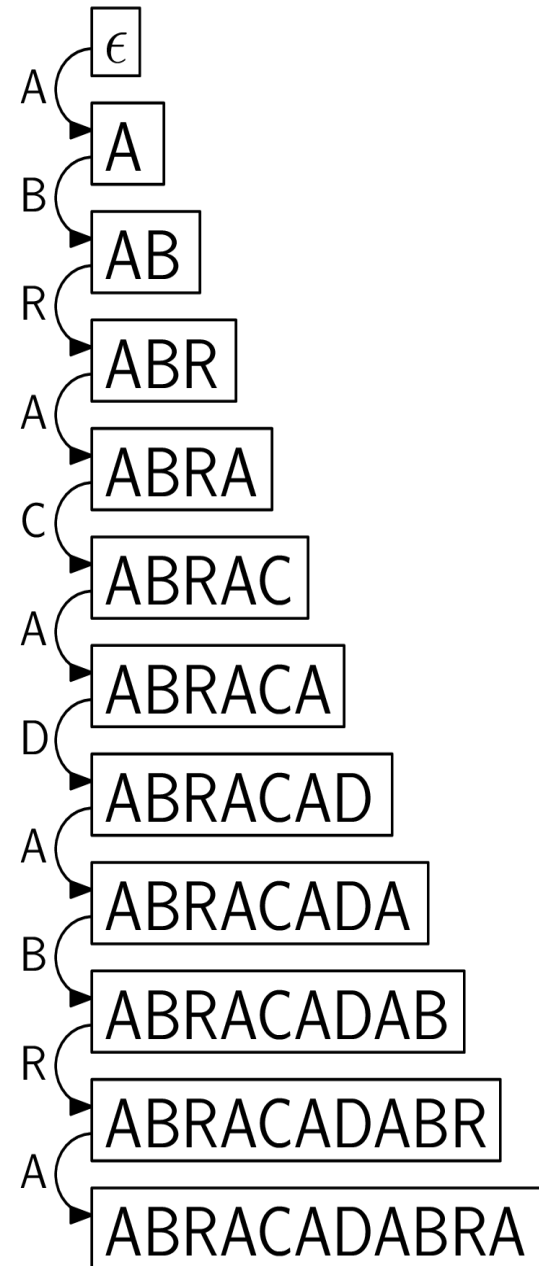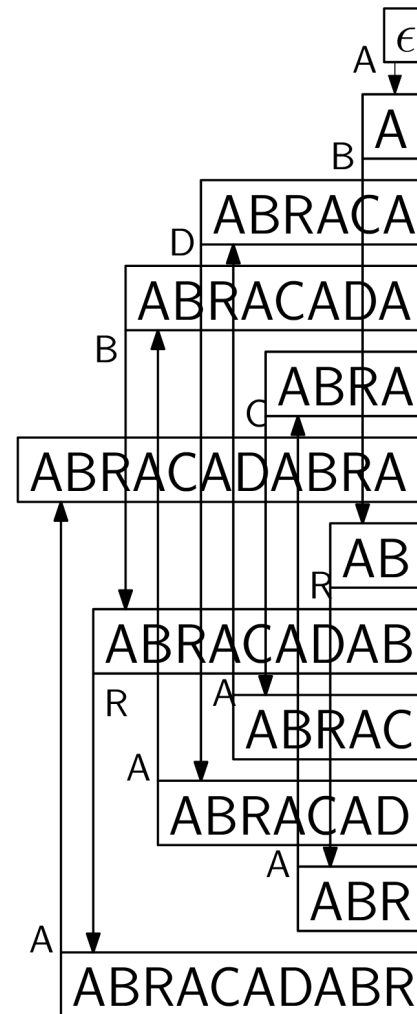# Search made simpler

Graph for
ABRACADABRA

Search made simpler

"Naturally" assign
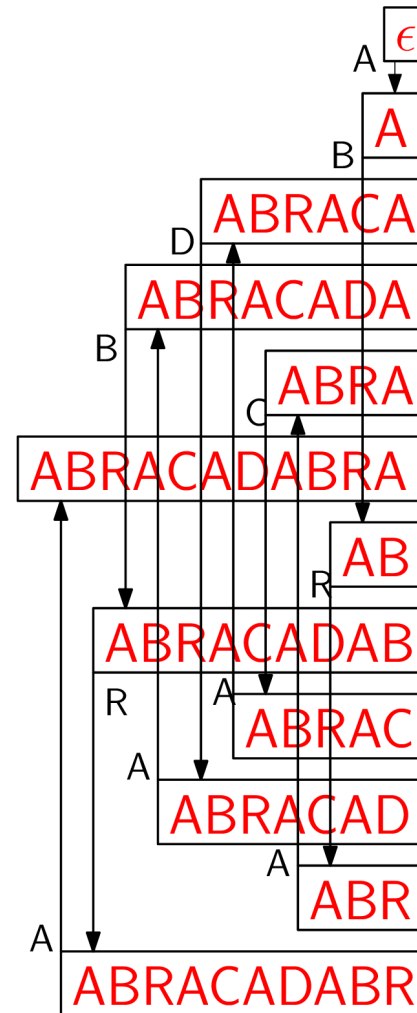a label to each node

Search made simpler

"Naturally" assign a label to each node
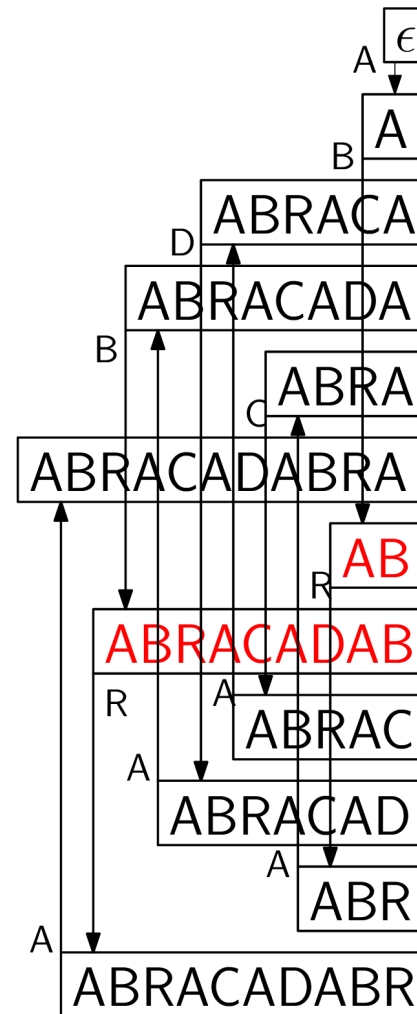
Arrange nodes according to labels
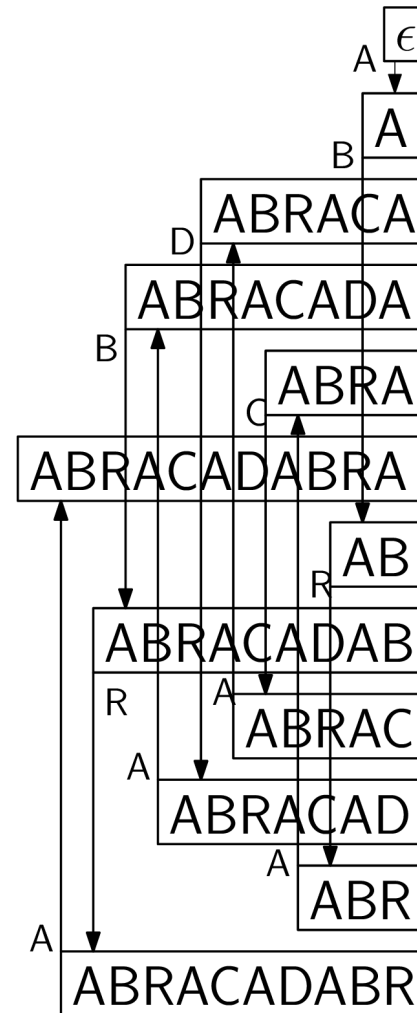
Searching in a sorted graph

Example:
Searching ABR

Searching in a sorted graph

Example:
Searching ABR

Searching in a sorted graph

Example:
Searching ABR

Searching in a sorted graph

Example:
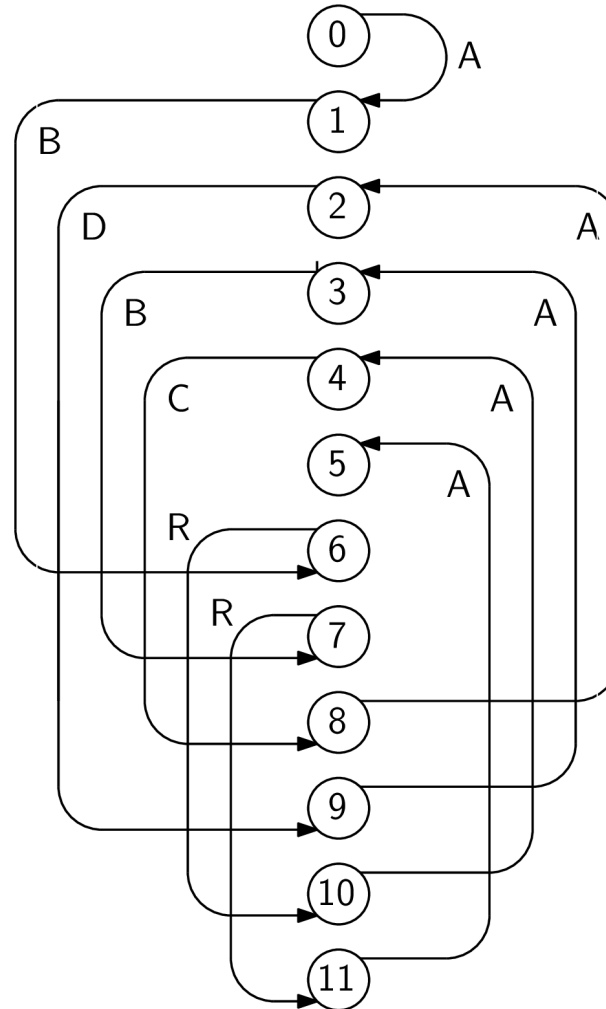Searching ABR

Two occurrences found!

Sorted graphs made simpler
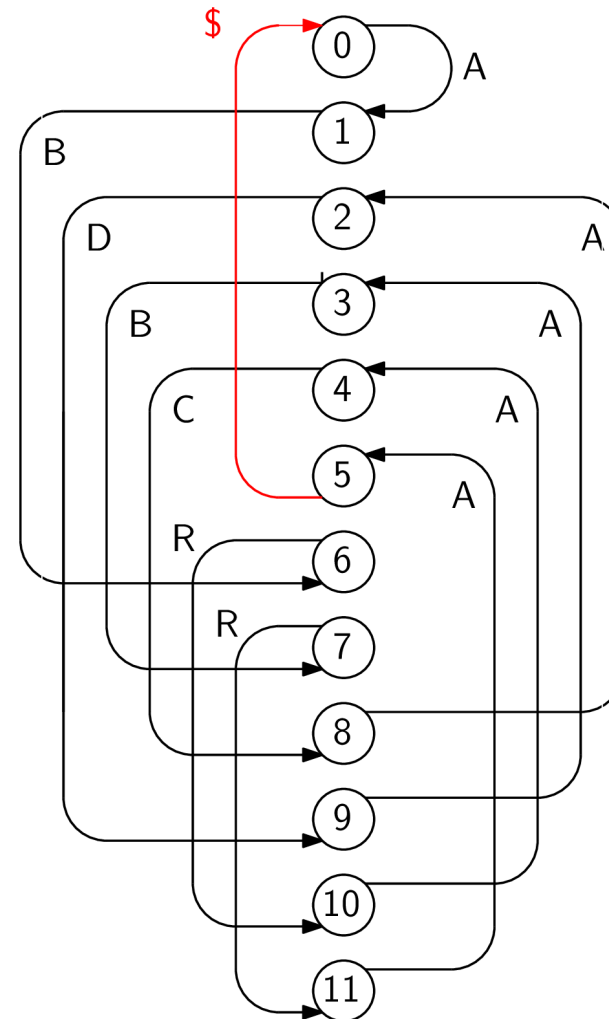
Easy to search sorted graph

Sorted graphs made simpler

No need to store node labels
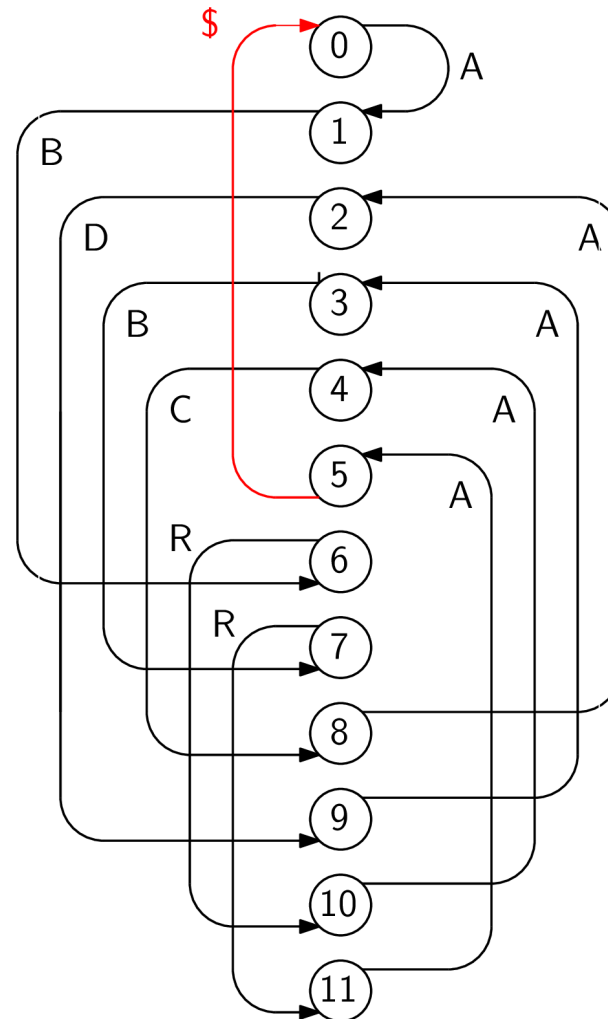
Sorted graphs made simpler

No need to store
the state labels

Add one arc
for symmetry

Sorted graphs made simpler

We identify nodes with outgoing labels

ABDBC$RRAAAA
is the BWT of
(ABRACADABRA$)^R

Search for subpaths in a linear graph can be simplified by rearranging nodes in a BWT-style

What about other graphs?