

Quantum Computing:

A Ridiculously Brief Motivation

Frederic Green

CS201

Clark University

Spring 2023

Here is a summary of the main differences between classical and quantum computing

We start with classical probabilistic computation.

Example: Miller-Rabin primality test. *Very roughly (!)* stated:

To determine if n is prime:


Randomly generate a bunch of numbers $m < n$.

Check that $m^k = 1 \pmod{n}$, or -1 , for certain (carefully chosen!) k .

If so, return TRUE, else FALSE.

As we generate more and more bits of m , the number of possible "configurations of memory" increases exponentially. But the right answer is obtained with very high probability.

A picture of this process:

time 

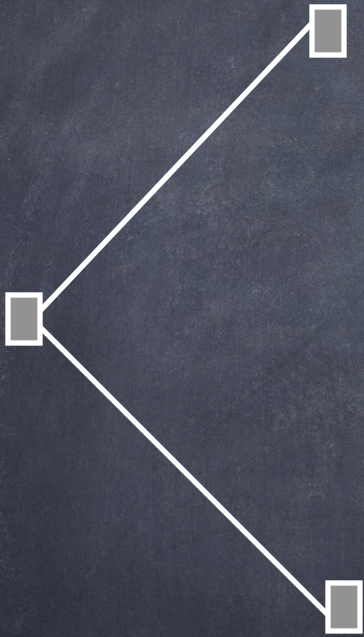


“possible configuration of
memory”

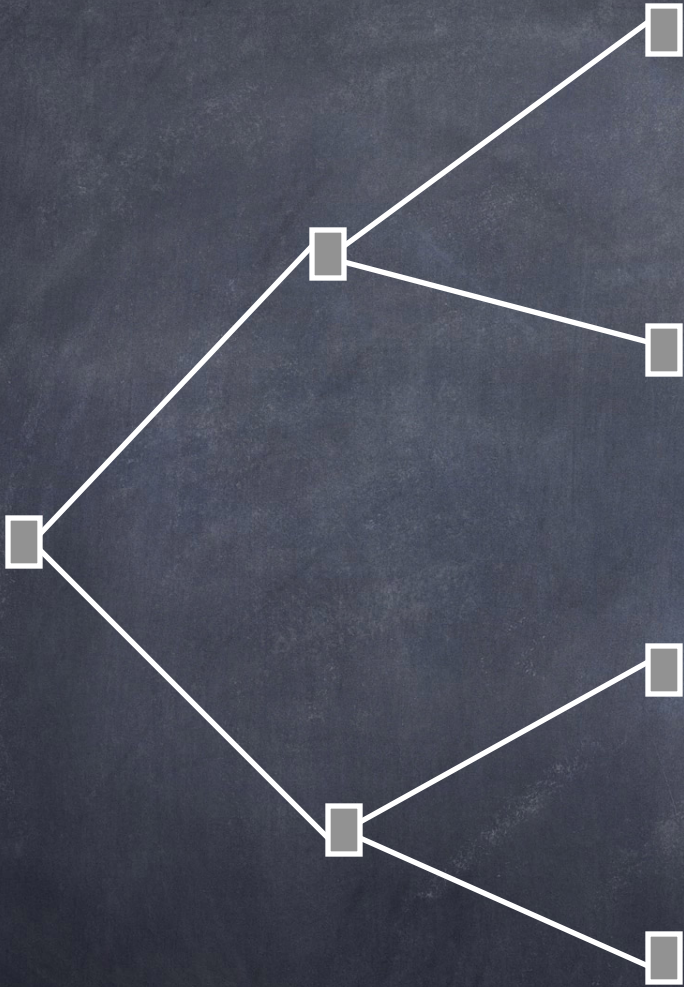
(only one when we start out)



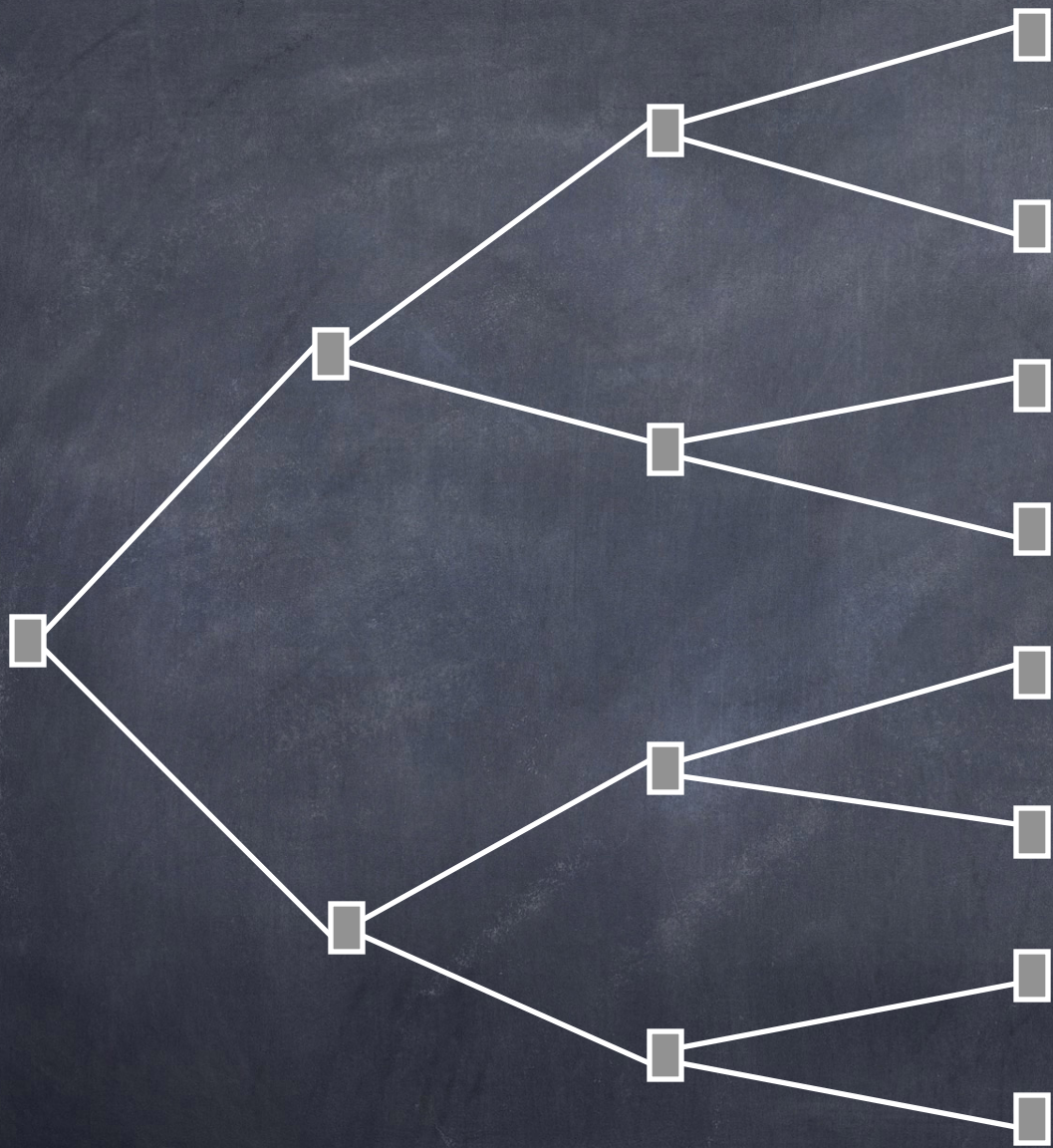
$t=0$



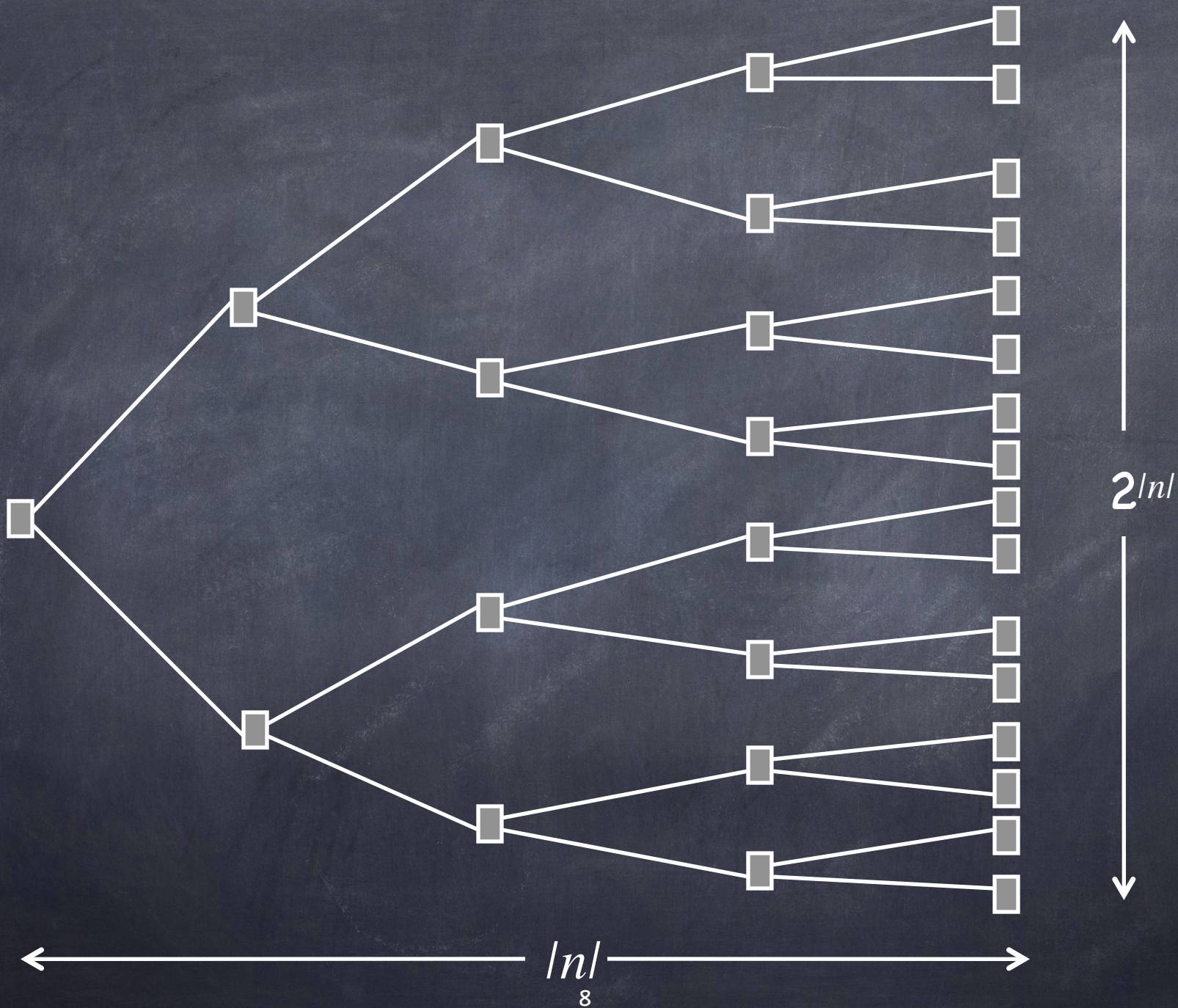
$t=1$



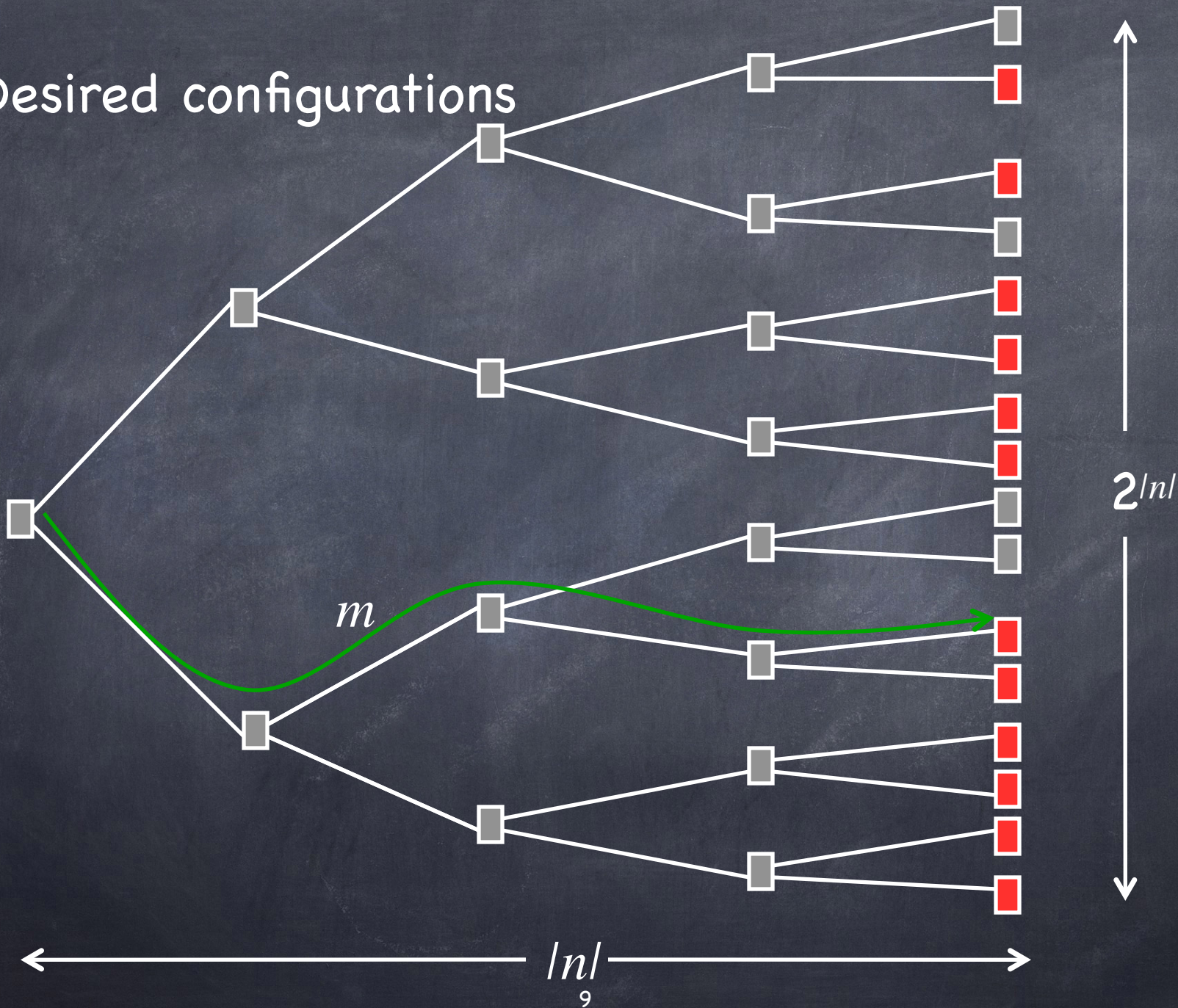
$t=2$



$t=3$



■ Desired configurations

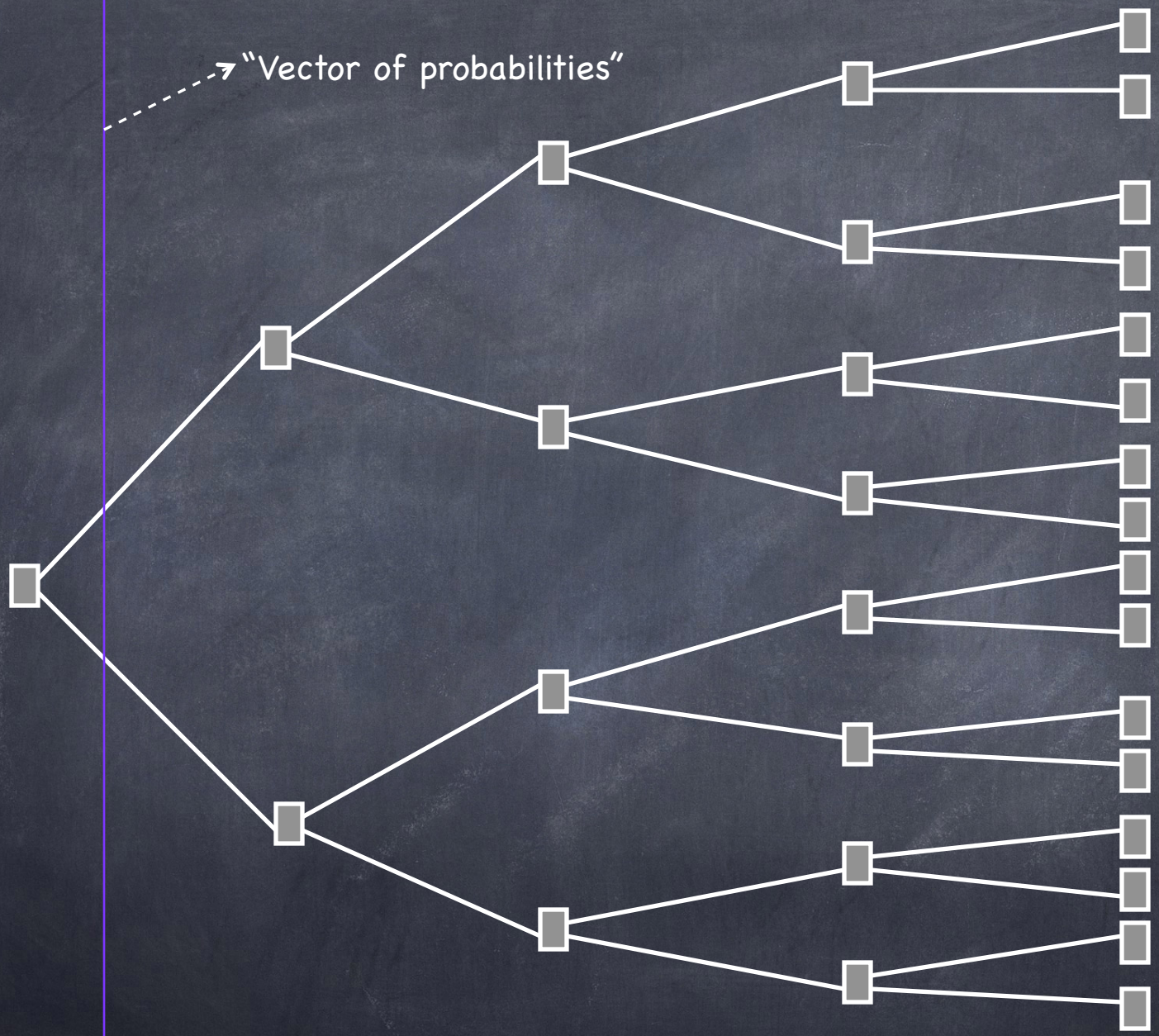


Interpretation:

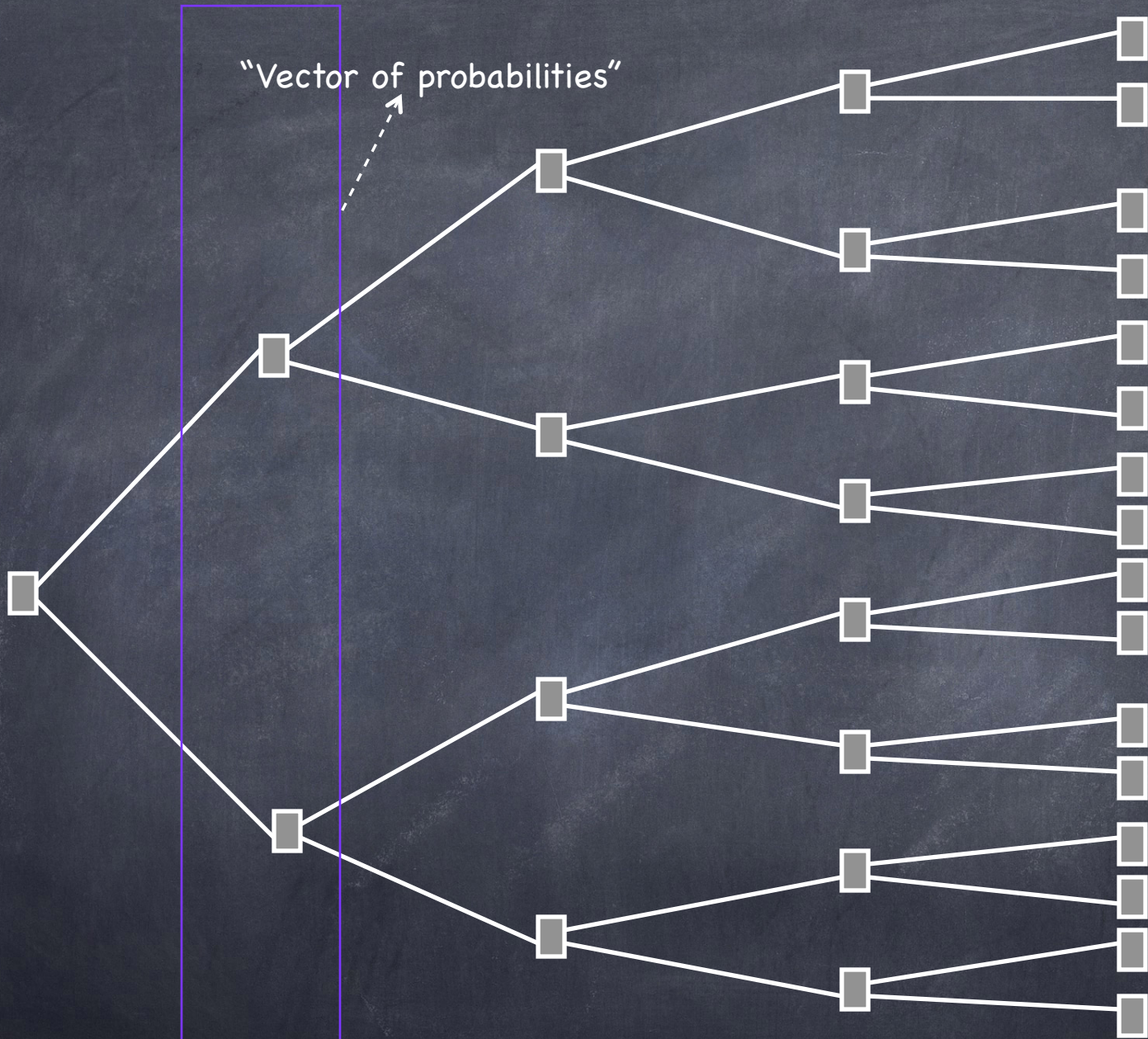
- At any step of the computation, memory can be in any one of a large number of configurations.
- Each configuration occurs with a certain probability; represent this set of probabilities as a vector (2^n components for n bits).
- The vector of probabilities evolves over time according to a certain set of rules determined by the algorithm.

$t=0$

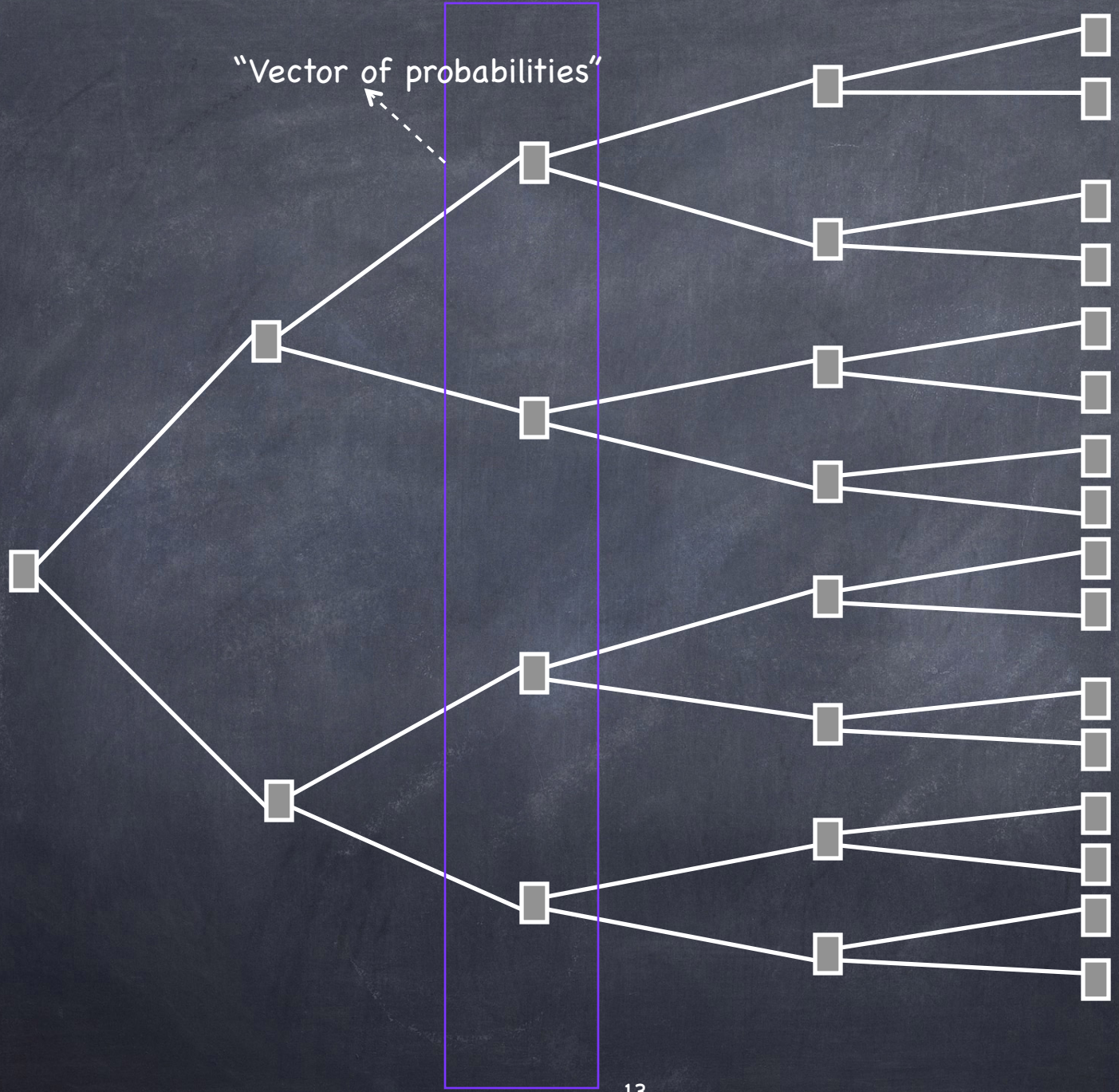
→ "Vector of probabilities"



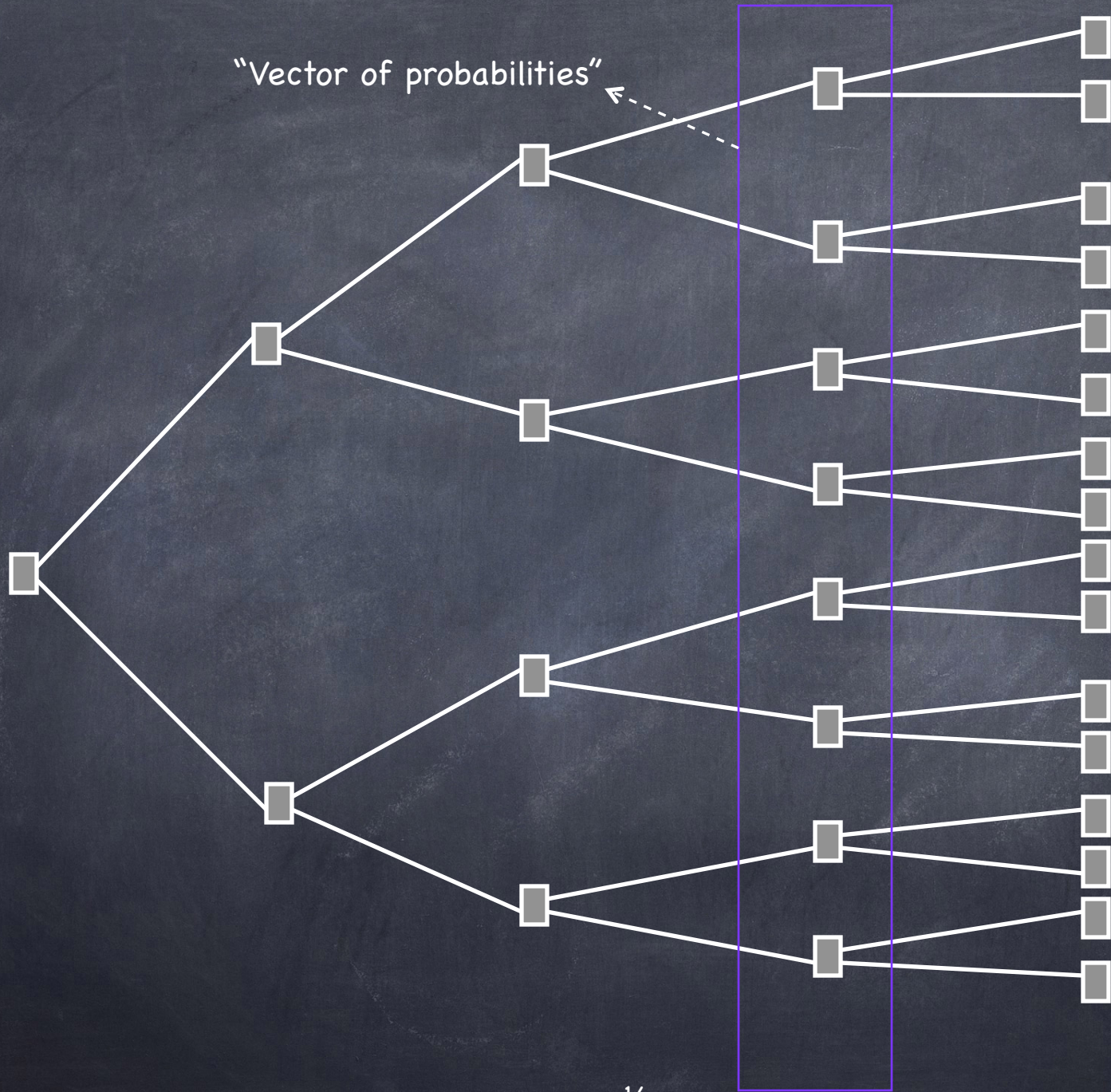
$t=1$



"Vector of probabilities"

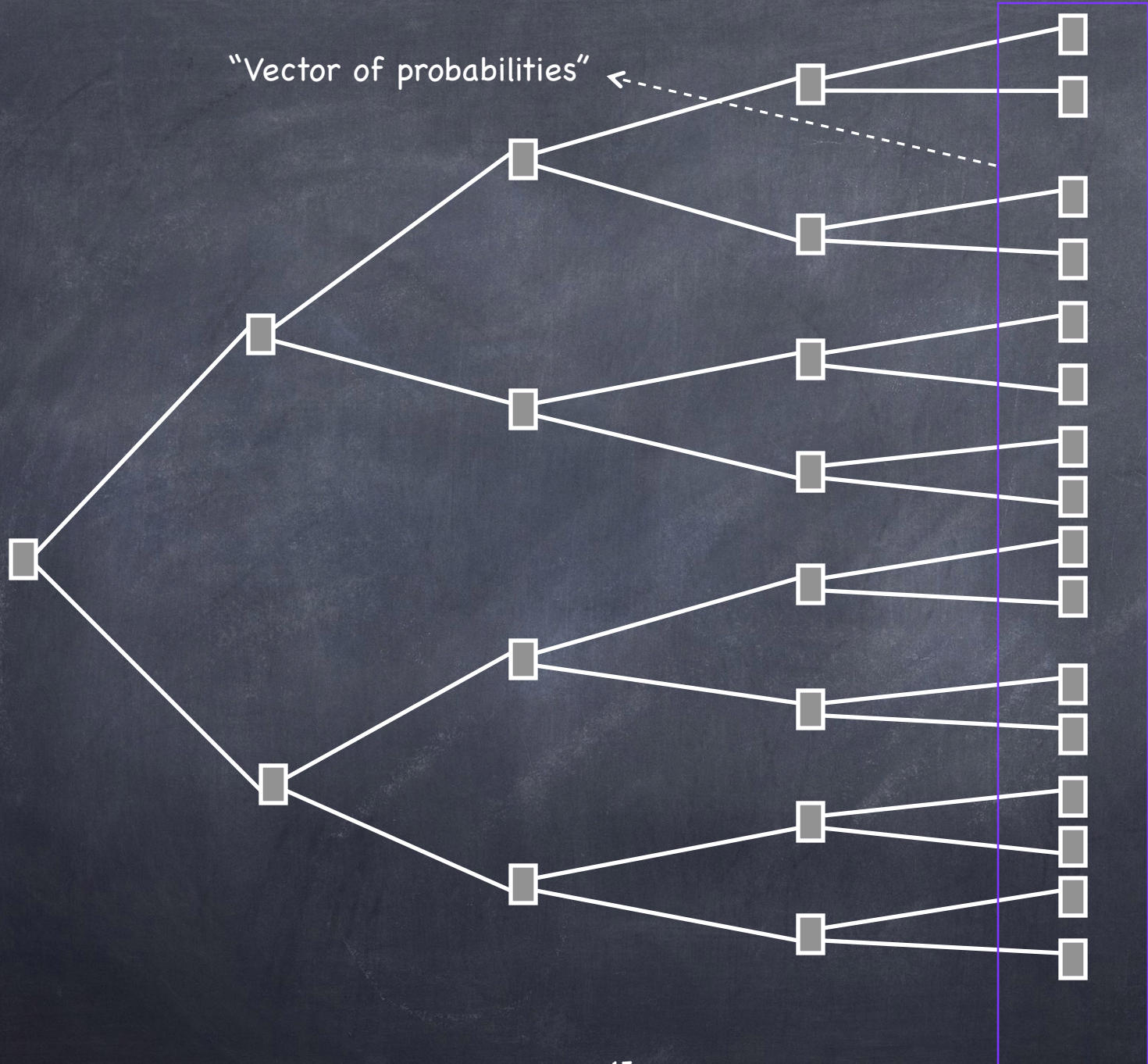


$t=2$



"Vector of probabilities"

$t=3$

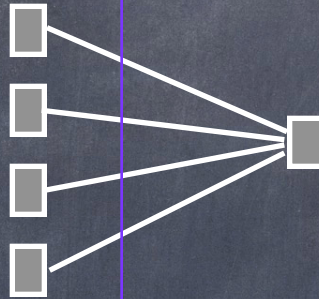


"Vector of probabilities" ←

$t=4$

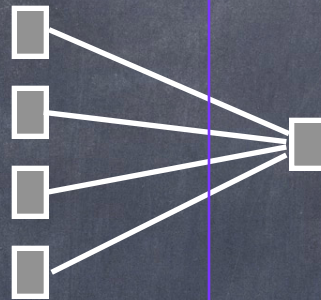
The "tree" representation is a little misleading:

many
configurations



$t=3$

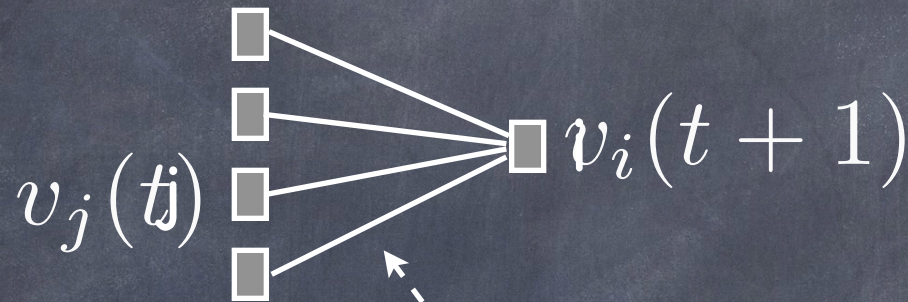
The "tree" representation is a little misleading:



can lead to one

$t=4$

...but this clarifies what's happening:



Let $M_{ij} = \text{prob that } j \text{ leads to } i$

$$\text{Then } v_i(t+1) = \sum_j M_{ij} v_j(t)$$

i.e., matrix multiplication!

.....to sum up:

Evolution of probabilities (classical formulation):

Let $v(t)$ = vector of probabilities at time t .

$v_i(t)$ = probability that we are in configuration i at time t .

We have seen that $v(t)$ evolves *linearly*.

I.e., there is a matrix M such that,

$$v(t+1) = Mv(t)$$

M_{ij} is simply the probability that configuration j yields configuration i .

So (of course!) M_{ij} is a real number in $[0,1]$.

M must leave $\sum_i v_i(t)$ invariant (prob's sum to 1).

Evolution of probabilities (*quantum* formulation):

Let $v(t)$ = vector of *probability amplitudes* at time t .
 $v_i(t)$ is a *complex number* whose square norm $|v_i(t)|^2 =$
probability that we are in configuration i at time t .

As in the classical case, $v(t)$ evolves *linearly*.

I.e., there is a matrix U such that,

$$v(t+1) = Uv(t)$$

U_{ij} is a *complex number* whose norm squared is the
probability that configuration j yields configuration i .

So (of course!) U_{ij} is *not necessarily* a real number
in $[0,1]$.

U must leave $\sum_i |v_i(t)|^2$ invariant (prob's sum to 1).

Hence U is unitary: $UU^\dagger = 1$.

Once again:

Evolution of probabilities (classical formulation):

Let $v(t)$ = vector of probabilities at time t .

$v_i(t)$ = probability that we are in configuration i at time t .

We have seen that $v(t)$ evolves *linearly*.

I.e., there is a matrix M such that,

$$v(t+1) = Mv(t)$$

M_{ij} is simply the probability that configuration j yields configuration i .

So (of course!) M_{ij} is a real number in $[0,1]$.

M must leave $\sum_i v_i(t)$ invariant (prob's sum to 1).

Evolution of probabilities (*quantum* formulation):

Let $v(t)$ = vector of *probability amplitudes* at time t .
 $v_i(t)$ is a *complex number* whose square norm $|v_i(t)|^2 =$
probability that we are in configuration i at time t .

As in the classical case, $v(t)$ evolves *linearly*.

I.e., there is a matrix U such that,

$$v(t+1) = Uv(t)$$

U_{ij} is a *complex number* whose norm squared is the
probability that configuration j yields configuration i .

So (of course!) U_{ij} is *not necessarily* a real number
in $[0,1]$.

U must leave $\sum_i |v_i(t)|^2$ invariant (prob's sum to 1).

Hence U is unitary: $UU^\dagger = 1$.

Measurement

After the computation has evolved, we may “measure” the configuration.

The matrix U , applied to the initial configuration, determines the probability that we end up in any given configuration.

Once the bits of the configuration have been measured, they will retain their measured value until acted on again (“collapse of the wavefunction”).

WHY?

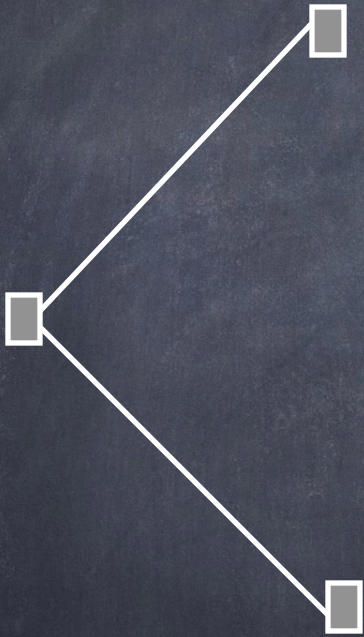
Don't ask!

Consequences

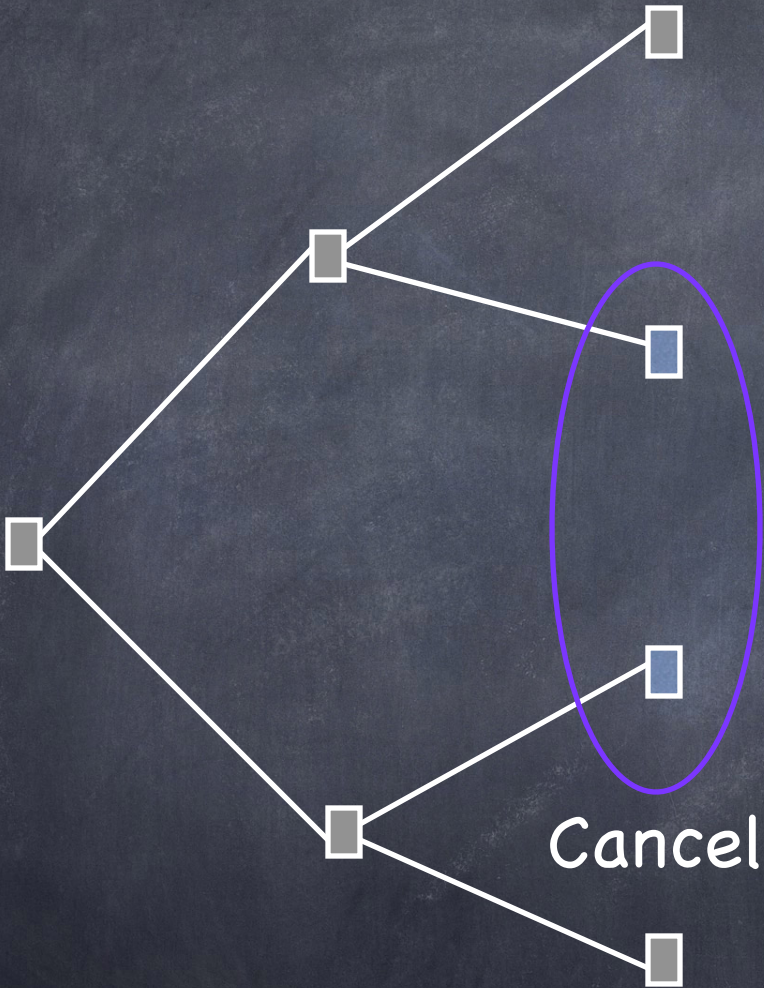
- Configurations can actually *cancel!*
- Because of unitarity, quantum computation is *reversible!*



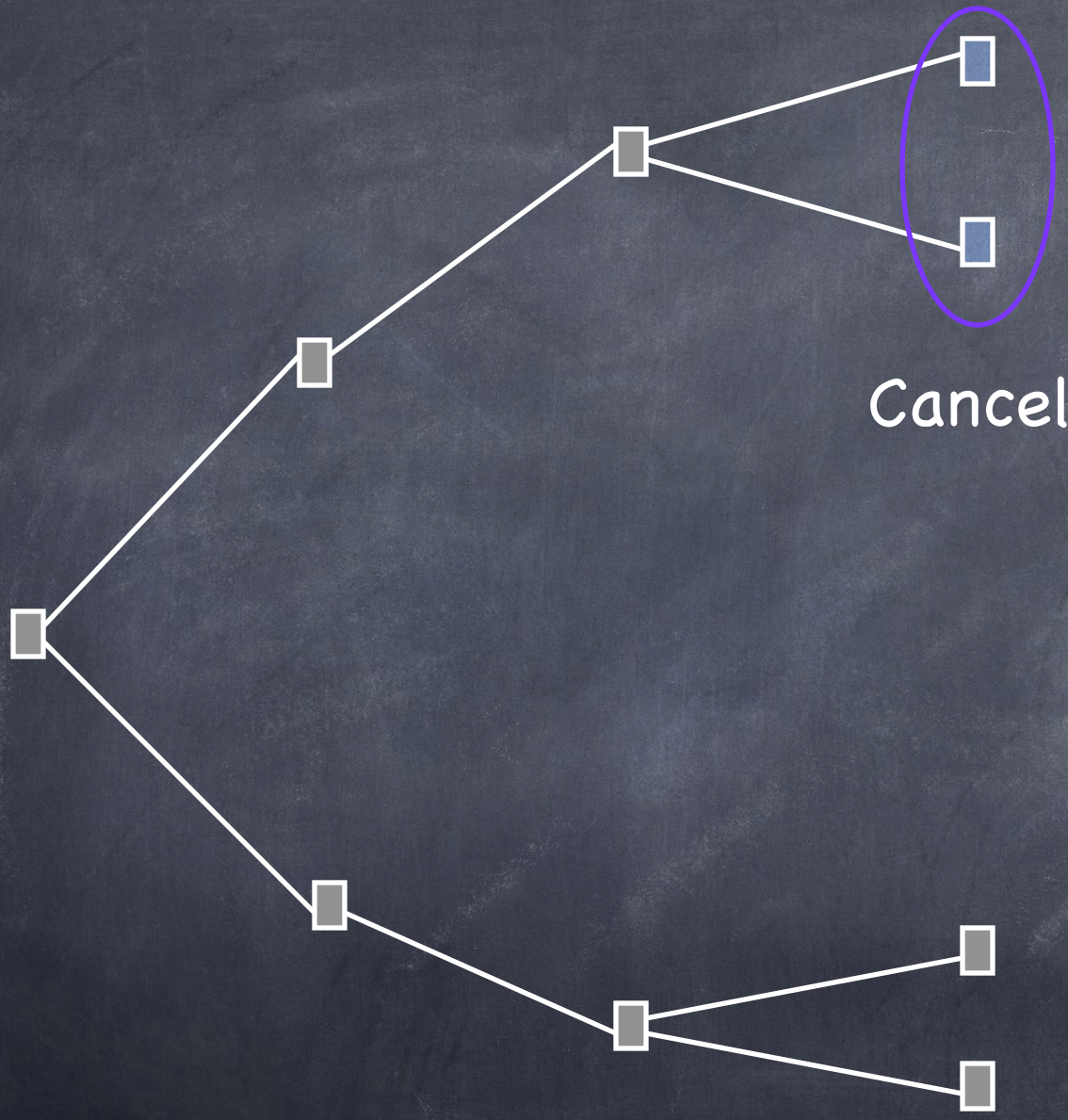
$t=0$



$t=1$



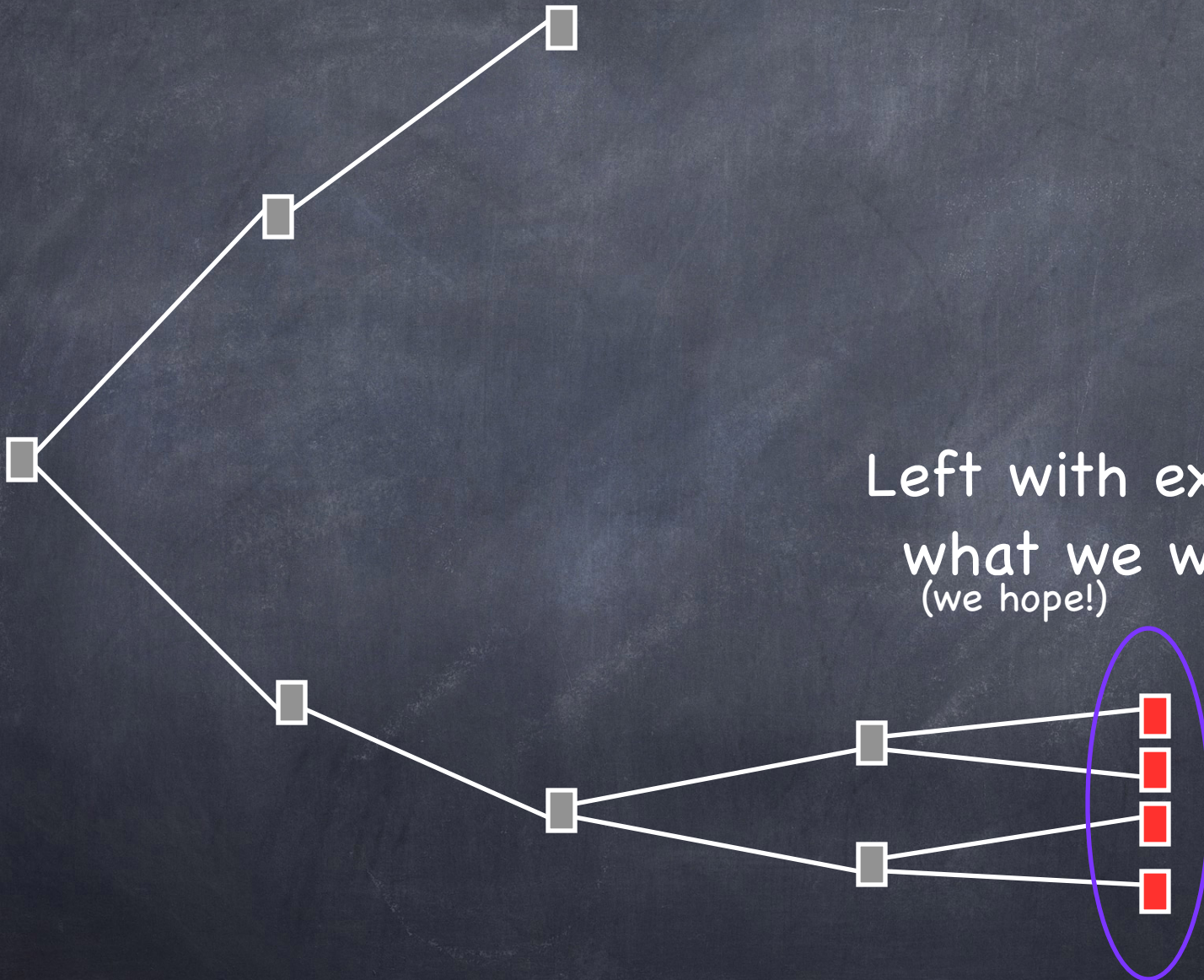
$t=2$



$t=3$



$t=4$



Left with exactly
what we want!
(we hope!)

$t=4$

Let's look at some of this stuff in more detail....