- Computation / algorithm – process of discrete steps (may be probabilistic)
- Input n-bit string
- $T(n)$ = # steps for any input of size n (worst case)
- Poly-time algorithm $T(n) = \mathcal{O}(\mathrm{poly}(n))$

Complexity classes of decision problems

P (poly time) – class of decision problems having deterministic poly-time algorithms

BPP (bounded error probabilistic poly time) – decision problems with probabilistic poly time algorithms, such that for every input Prob(answer is correct) $\geq 2/3$.

Fact: can replace $2/3$ by any constant $\frac{1}{2} + \delta$, $0 < \delta < \frac{1}{2}$ and BPP same

If have $\frac{1}{2} + \delta$ algorithm (δ small) repeat K times, take majority vote as answer. Chernoff bound prob(majority vote correct) $> 1 - e^{-2\delta^2 K}$ so can be $>$ any $1-\epsilon$ (ϵ small) for suitable const K and $K \times$ poly is poly.

BPP ~ classically feasible computations, computable in practice ~ poly time, tolerate small error

Example Primality testing for N, input size $\log_2 N$

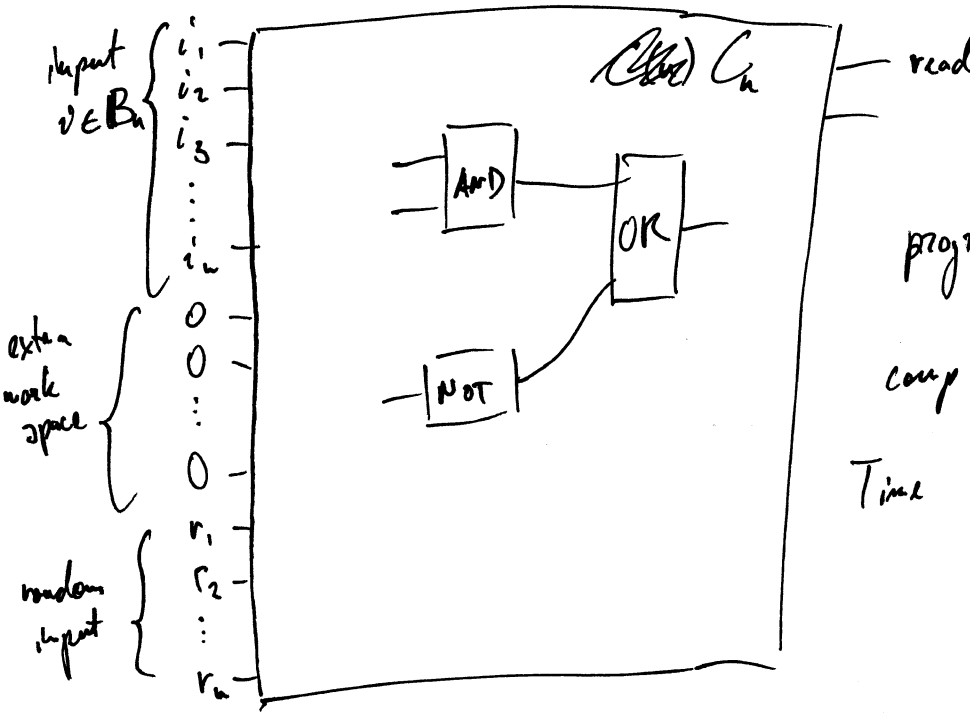
- naive test divide N by $1, 2, \ldots, \sqrt{N}$? not poly time
  need $\sqrt{N}$ trial divisions $2^{\frac{1}{2}\log N} = 2^{\frac{1}{2}n}$
- choose random $k < N$ & test divide N/k? poly time probabilistic but (probability ans correct ~~is~~ not $> \frac{1}{2}$)

- known to be in BPP (~1976)
- known to be in P (2004)

We will use circuit model of computation

Classically

For each input size n have a prescribed circuit of Boolean AND/OR/NOT gates



program in machine language

comp steps are the gates

Time $T(n)$ = size of the circuit C_n
$\equiv$ total number of gates

For full computation need circuit family $\equiv$ algorithm $C_1, C_2, \ldots, C_n, \ldots$

Universal set of gates G can make any Boolean $f: B_m \to B_n$ as a circuit of gates from G

<u>Quantum computation - circuit model</u>

For input $x = i_1 \ldots i_n$ start with qubits $|i_1\rangle|i_2\rangle \ldots |i_n\rangle|0\rangle \ldots |0\rangle$ (random bits not needed)

Now computational steps (gates) are quantum gates = unitary operations on designated (few) qubits

Basic unitary gates commonly used

$$X, \; Z, \; H, \; P_\varphi = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{pmatrix}, \; CX, \; CZ$$

Single qubit U is 2×2 matrix.

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

$H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

$CX|i\rangle|j\rangle = |i\rangle X^i |j\rangle$