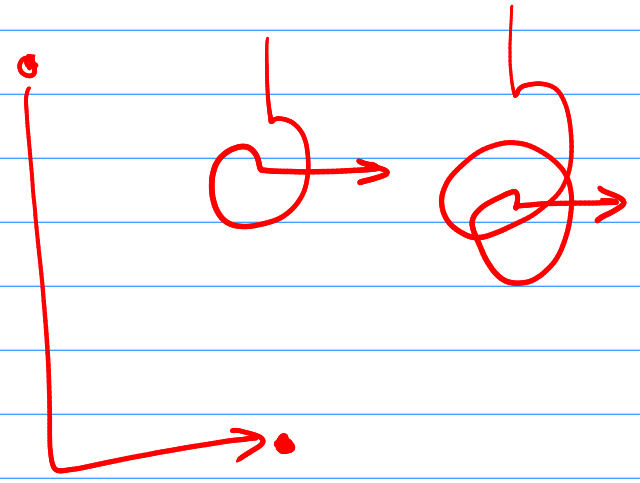
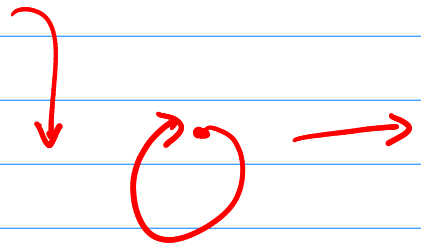
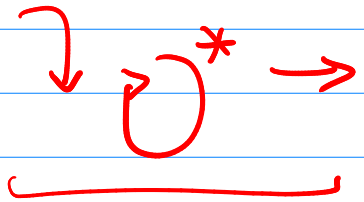
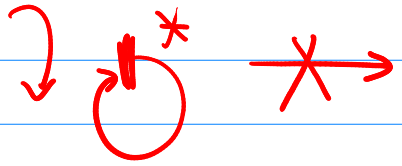
```
countBy2(n):
    i := 1   j := 2
    while j < n
        if i ≥ n : crash
        i := i+1
        j := j+2
```



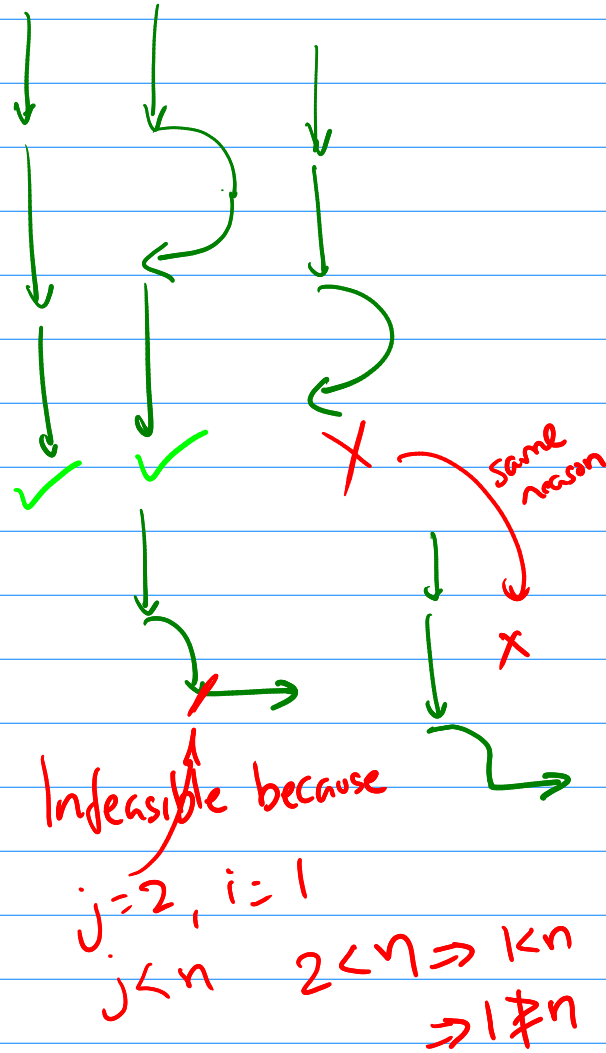


TST: All paths of the form  are infeasible.



Let's forget about loops!

```
count2-Fin(n):
    i := 1    j := 2
    if j < n
        if i ≥ n : crash
        i := i+1    j := j+2
    if j < n
        if i ≥ n : crash
        i := i+1    j := j+2
```



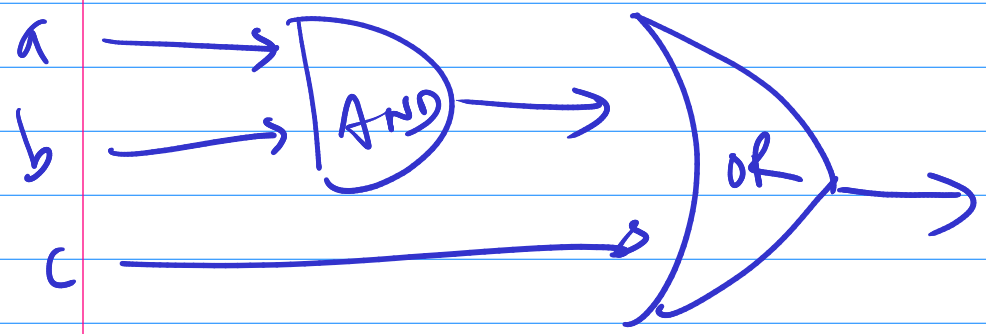
Still too many paths

Straight Line Code

```
countBy2Straight(n)
    i := 1    j := 2
    assert (j ≥ n or i < n)
```

Is this path feasible?

Stateless (Combinatorial) Electrical Circuits



Does this circuit ever produce the value 1?

"feasible"?

Propositional Logic

- Propositions : a b c d e

a: It is raining

b: Sam is wet

$(a \Rightarrow b) \wedge \neg b \wedge a$

- Connectives

and or not $a \wedge b$ $a \vee b$ $\bar{a}$

$\Rightarrow$ nand nor

- What do these connectives "mean"?

Truth tables

a	b	$a \wedge b$	$a \vee b$	$\bar{a}$
T	T	T	T	F
T	F	F	T	F
F	T	F	T	T
F	F	F	F	T

← Each of these rows is called a model.

Proof systems / Proof rules

$(a \Rightarrow b) \quad a \quad \vdash b$

Modus ponens

$(a \Rightarrow b) \quad \neg b \quad \vdash \neg a$

Modus tollens

$a \wedge b \vdash a$

$a \wedge b \vdash b$

$a \vdash a \vee b$

$b \vdash a \vee b$

Truth — Proof

Expressiveness — Models

Definability

Soundness

Completeness

Gödel's Completeness Thm

Gödel's Incompleteness Thm

Q1. Can you prove everything that is true?

Q2. Is everything that you can prove really true?

Algorithmic Questions

$$(a \wedge b \Rightarrow c) \wedge (c \wedge b \Rightarrow a) \wedge \bar{a} \wedge b$$

a	b	c	$a \wedge b \Rightarrow c$	$c \wedge b \Rightarrow a$	Final
T	T	T	T	T	F
T	T	F	F	T	F
T	F	T	T	T	F
T	F	F	T	T	F
F	T	T	T	F	F
F	T	F	T	T	T
F	F	T	T	T	F
F	F	F	T	T	F

Satisfiability — NP-complete

Q1. Does the truth table have a T?

Q2. Does the truth table have an F?

Q2. Does the truth table only consist of Ts?

Validity — coNP-complete problem

Q3. How many Ts?

Model counting

#P-complete

Counting SAT

Claim φ is sat $\overset{\times}{\Rightarrow}$ φ is valid

φ is sat $\overset{\checkmark}{\Leftarrow}$ φ is valid

Grammar of Boolean Formulas

$$\varphi ::= a \mid b \mid c \mid \cdots$$

$$\mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \neg\varphi$$

Conjunctive Normal Form

$\varphi = ($ ___ or ___ or ___ $)$ and

$($ ___ or $\neg$ ___ $)$ and

$($ ___ or $\neg$ ___ or ___ $)$

clause

literal

Literal : $a \mid \bar{a}$

Clause: $Lit_1 \vee Lit_2 \vee \cdots \vee Lit_k$

Each clause has at most 3 literals

1-CNF 2-CNF 3-CNF 4-CNF 5-CNF

NP hard

Checking SAT is trivial

SAT can be checked in linear time

Checking SAT is NP-complete

Checking validity is trivial.

$(a \vee b) \wedge (\neg b \vee c) \wedge (b \vee d)$

a is false

Disjunctive Normal Form

$\varphi \equiv$ (___ and ___ and ___) or

(___ and ¬ ___) or

(¬ ___ and ¬ ___ and ___)

Term

DNF-SAT : Trivial (Linear time)

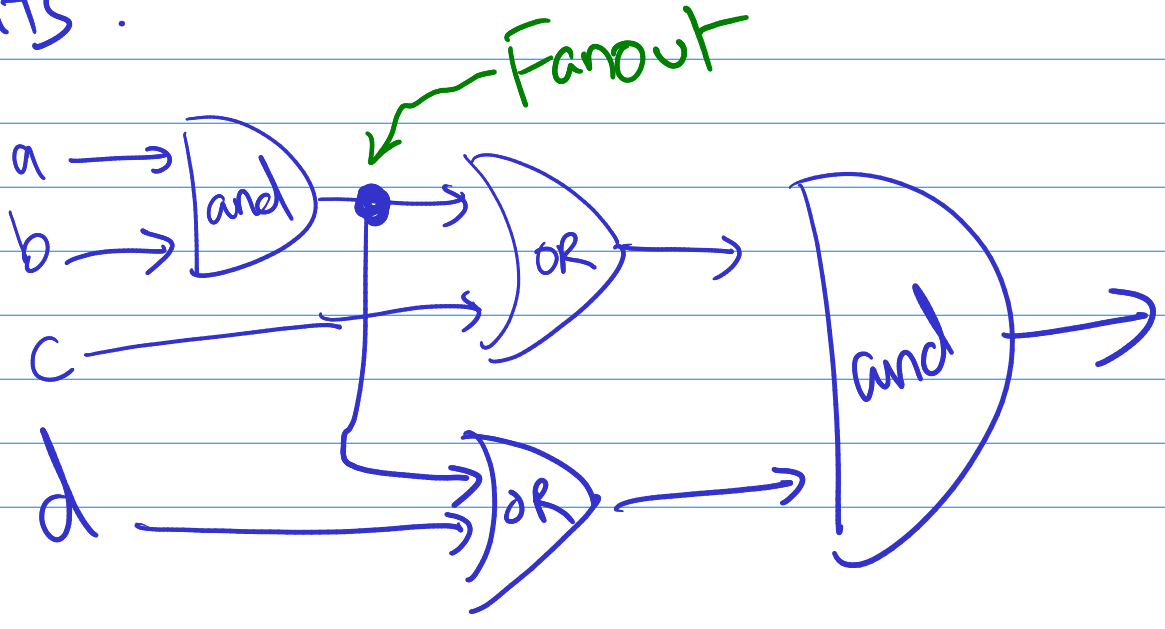
DNF-Validity : coNP-complete

Moral: Formula representation matters

	General Boolean Formulas	CNF	DNF	Circuits	ROBDD
SAT	NPC	NPC	Linear	NPC	Linear
Validity	coNP-c	Linear time	coNP-c	coNPC	Linear
MC	#PC	#P-c	#P-c	#PC	simple polynomial

Formulas: $(a \wedge b \Rightarrow c) \wedge (c \wedge b \Rightarrow a) \wedge \bar{a} \wedge b$

Circuits:



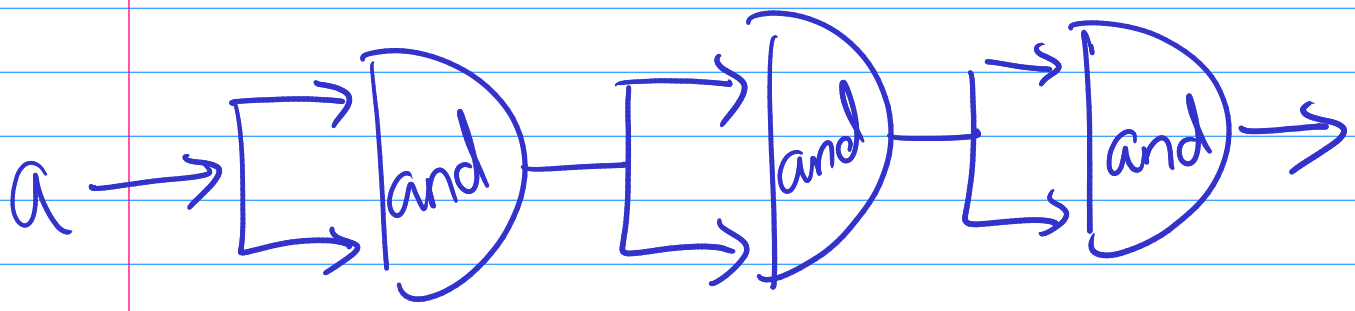
$$((a \wedge b) \vee c) \wedge ((a \wedge b) \vee d)$$

Evaluated twice

For us: Fanout = variable bindings

$$x = \text{factorial}(n)$$

$$x \; ? \; x * x$$



$$((a \wedge a) \wedge (a \wedge a)) \wedge ((a \wedge a) \wedge (a \wedge a))$$

Circuits are exponentially more succinct than formulas.