# CSE322 Theory of Comput. (L24)

naive complexity $O(n^3)$

$3S = \{ \langle$Integer array $A$, integer $T\rangle$

: $A$ contains a 3-element subarray

that sums to $T \}$

$SS = \{ \langle$Integer array $A$, integer $T\rangle$ naive complexity $O(n 2^n)$

: $A$ contains a subarray that sums to $T \}$

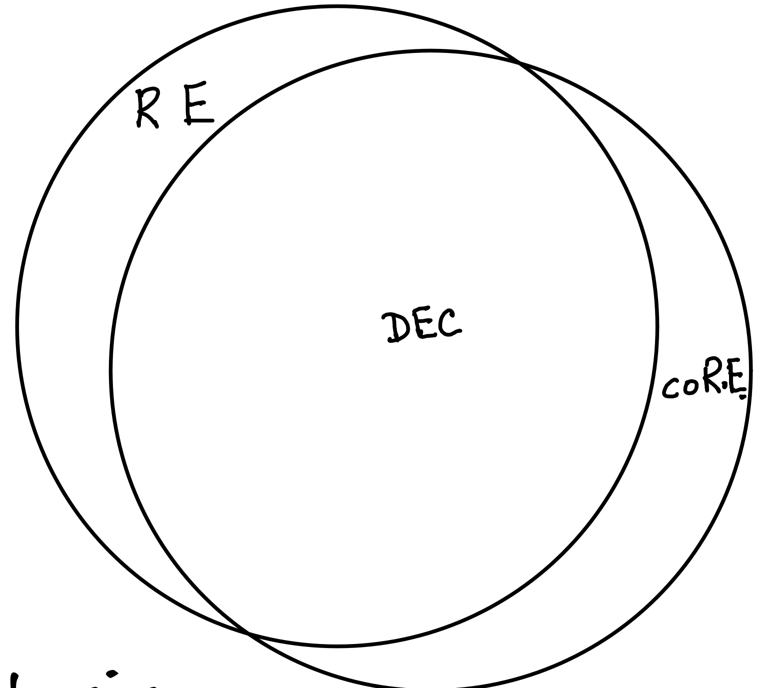
# Subclasses of DEC

$L = \{ w.\mathrm{rev}(w) : w$ is a binary string $\}$

$L = \{ w : w$ represents 3-colourable graphs $\}$

L is decidable, but in which subclass?

Subclasses of decidable languages




# Worst-case Running-time Analysis

1. M : fixed model of computation
2. Define: Running "time" of M on inputs

For problem P, Algo-A is better than Algo-B if

... for every input, Algo-A finishes before Algo-B

... the maximum time by Algo-A (on any input) is lesser than that of Algo-B (on any input)

"maximum time" is not well-defined for arbitrary length

# Time complexity of TM

M : 1-tape DTM

$T(M,w)$ = Number of steps needed by M on input w

M has complexity $f(n) : \mathbb{N} \to \cancel{\mathbb{R}^+}\ \mathbb{N}$

$\star\ f(n) = \max_{(w:\ |w|=n)} \{ T(M,w) \}$

Q: How to define average and randomized complexity?

| | | |
|---|---|---|
| ε | 21 | $f(0)=21$ |
| 0 | 15 | |
| 1 | 73 | $f(1)=73$ |
| 00 | 61 | |
| 01 | 23 | |
| 10 | 110 | $f(2)=110$ |
| 11 | 99 | |
| ⋮ | | ⋮ |
| | | $f(n)= \ldots$ |

**For NDTM:**

$T(M,w)$ = max. #steps needed by M(w) on any "non-deterministic branch"

$7n^3 + 15n + 19$

Define: TM M has time-complexity f(n) if f(n) is the maximum number of steps used by M on any input of length n.

Define: L has time complexity O(f(n)) if there is some TM M deciding L with complexity f(n).

$O(7n^3 + 15n + 19) = O(n^3)$

$7n^3 + 15n + 19$

# Complexity classes

$t(n) = n^3 + 2n$

DTIME(n) = { L : L can be decided in O(n) by some TM }

(D)TIME(t(n)) = { L : there is a (D)TM M that decides L and runs in time O(t(n)) }

Why O(...) ?

DTIME(n) ⊆ DTIME($n^2$)

L ∈ DTIME(n)

Similarly, define NTIME(t(n)).

If L is in DTIME(t(n)) then L is said to have complexity O(t(n)).

L ∈ DTIME(?)

What is the complexity of L = { w.rev(w) } ?

∈ DTIME($n^2$)

What is the complexity of L = {$0^n 1^n$ : n >= 0 }

Observe dependence on M.

(more about it later)

L ∈ DTIME(n)

To show L ∈ DTIME($n^2$)

$f(n) \in O(n^2) \Rightarrow \exists c, n_0$ s.t.

$\forall n \geq n_0$,

$f(n) \leq c \cdot n^2$

$\Rightarrow f(n) \in O(n^2)$

All DTIME(f(n)) classes within decidable.

# Complexity class: P (polynomial)

$P = \bigcup_{k \geq 0} DTIME(n^k) = \cancel{DTIME(n^{100})}$

$P(k) = TIME(n^k)$

$P = P(0) \cup P(1) \cup P(2) \cup \ldots$

$= DTIME(n) \cup DTIME(n^2) \cup DTIME(n^3) \cup \ldots$

For any L in P, L can be solved (decided) by a 1-tape DTM in $\underline{\quad O(n^k) \quad}$

$L \in DTIME(n^k)$ for some $k$

$EXP(k) = TIME(2^{n^k})$

$EXP = EXP(0) \cup EXP(1) \cup EXP(2) \cup \ldots$

$EXP = \bigcup_{k \geq 0} DTIME(2^{n^k})$

For any L in EXP, L can be solved (decided) by a 1-tape DTM in ________

Exercise

Q? P is class of L such that there exists a DTM M such that …

for every n,

there exists a polynomial p(n) such that

for every w of length n, M can decide w in at most p(n) steps.

# Change of model

If 2-tape M took $k(n)$ steps,
then its 1-tape version would take $k^2(n)$ steps.

Thm. Let $t(n) >= n$.
If L is in TIME(t(n)) using k-tape DTMs,
then L is in TIME(t(n) * t(n)) using 1-tape DTMs
(even better, TIME(t(n) * log(t(n))) using 1-tape DTMs).

Lemma. If L is in P using k-tape DTMs, then L is in P using 1-tape DTMs.

If $L \in DTIME^{2tape}(n)$
then $L \in DTIME^{1tape}(n^2)$

$P^{2tape} = \bigcup_{k \geq 0} DTIME^{2tape}(n^k)$

$DTIME^{2tape}(n^k) \subseteq DTIME^{1tape}(n^{2k})$

For classes like P and larger ... use multi-tape TMs.
For "smaller classes below P", define carefully.

# NP (non-deterministic polynomial)

(recall) NTIME(f(n)) = ? $\{L : L$ can be decided by an NDTM in $O(f(n))$ steps$\}$

NP(k) = ? $\exists k$ s.t. HAMCYCLE $\in$ NTIME($n^k$)

NP = ? $\bigcup_{k \geq 0}$ NTIME($n^k$)

Thm: HAMCYCLE is in NP. = $\{\langle G\rangle :$ G has a cycle that visits every vertex exactly once$\}$

Proof: Construct an NDTM that runs in O(poly(n)) and decides HAMCYCLE.

def NDTM($\langle G\rangle$):
n = # vertices in G
non-deterministically write n vertices $v_1 \dots v_n$ $O(n^2)$
if every $v_i$ is distinct & $\forall i, (v_i, v_{i+1}) \in E$ & $(v_n, v_1) \in E$, then goto $q_{accept}$. $O(n^2)$
else goto $q_{rej}$.

Claim: NDTM decides the.

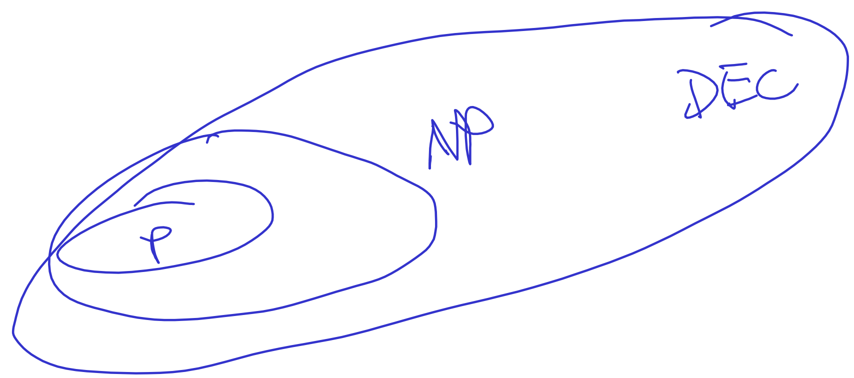
Claim: $O(n^2)$ for every non-det. branch.

Show that SORTED = { <integer array A : A is sorted in the incr. order } is in NP.

$DTIME(t(n)) \subseteq NTIME(t(n))$

If L can be decided in $O(t(n))$ by a DTM

then " " " " $O(t(n))$ " " NDTM.



$P \subseteq NP \subseteq EXP$


Exercise

Show that P is a subset of NP

Show that NP is a subset of EXP

$NTIME(n^k) \subseteq DTIME(2^{n^k} \ldots)$

Q: Are there decidable problems that are not in P?

Q: Are there decidable problems that are not in NP?

Q: Are there decidable problems that are not in EXP?

Q: Are there problems in NP that are not in P?

Q: Are there problems in EXP that are not in NP?

Q: Are there problems in EXP that are not in P?