

Tentamen IN3105 Complexiteitstheorie

16 april 2012,

9.00 – 12.00 uur

- Dit tentamen bestaat uit 10 meerkeuzevragen, 5 korte (open) vragen en 2 open vragen.
- Per meerkeuzevraag kunnen 0 tot 4 alternatieven juist zijn.
- Voor de meerkeuzevragen kunt u maximaal 50 punten behalen, voor de korte open vragen maximaal 30 punten en voor de open vragen maximaal 20 punten.
- Het totaal aantal punten voor de meerkeuzevragen (mp) wordt als volgt bepaald:
 - o per meerkeuzevraag i wordt het aantal foutief beantwoorde alternatieven f_i bepaald (0-4);
 - o de som $f = \sum f_i$ voor alle 10 meerkeuzevragen wordt bepaald ($0 \leq f \leq 40$);
 - o het aantal punten mp is dan gelijk aan $mp = 50 \times \max\{0, (40 - 1.5 \times f) / 40\}$.
- Het eindcijfer c wordt bepaald door de som te nemen van mp , het totaal aantal punten behaald voor de 5 korte vragen kv en de som van het aantal punten behaald voor de twee open vragen ov , en vervolgens het resultaat te delen door 10: $cv = (mv + kv + ov) / 10$; dit cijfer wordt vervolgens afgerond op het dichtstbijzijnde halve of gehele getal.
- Het gebruik van dictaat, aantekeningen, boek of andere bronnen is niet toegestaan.
- Het gebruik van grafische en niet-grafische rekenmachines is eveneens niet toegestaan.
- Beantwoord de meerkeuze vragen op het bijgeleverde antwoordformulier.
- Beantwoord de open vragen zo bondig mogelijk; *geef geen irrelevante informatie, dit kan leiden tot puntenaftrek!*
- Controleer of u op ieder blaadje uw naam en studienummer heeft vermeld en geef het totaal aantal ingeleverde bladen op (tenminste) de eerste pagina.

Notationele conventies:

- $A \leq B$ betekent dat er een polynomiale-tijd reductie van A naar B bestaat.
- A^c is het complement van het probleem A .
- Er wordt steeds vanuit gegaan, dat $P \neq NP$ en $NP \neq coNP$, tenzij uitdrukkelijk het tegendeel wordt vermeld.

MEERKEUZEVRAGEN (50 PUNTEN)

Vraag 1. Voor elk probleem $A \in \text{NP}$ geldt:

- a. er is een algoritme dat A beslist in polynomiale ruimte en exponentiële tijd.
- b. $B \leq A$ voor elk NPC probleem B .
- c. $A \leq B$ voor $B \in \text{PSPACE}$.
- d. $X \leq A$ en $A \leq Y$ als $X \in \text{P}$ en $Y \in \text{NP-hard}$.

Vraag 2. Gegeven zijn de volgende drie problemen:

- | | |
|--------------|--|
| CONNECT | Gegeven een graaf $G = (V, E)$, is er een pad tussen elk paar knopen in V ? |
| NONHAMP | Gegeven een graaf $G = (V, E)$, heeft G geen hamiltons pad? |
| VERTEX COVER | Gegeven een graaf $G = (V, E)$ en een integer K , heeft G een vertex cover ter grootte van K ? |

Er geldt:

- a. CONNECT is een **P**-probleem.
- b. VERTEX COVER kan worden beslist door een polynomiaal algoritme.
- c. $\text{CONNECT} \leq \text{VERTEX COVER}$.
- d. $\text{VERTEX COVER} \leq \text{NONHAMP}$.

Vraag 3. Welke van de onderstaande inclusierelaties is/zijn geldig:

- a. $\text{NPSPACE} \subseteq \text{PSPACE}$
- b. $\text{P}^{\text{NP}} \subseteq \text{NP}^{\text{P}}$
- c. $\text{NPC} \subseteq \text{PP}$
- d. $\text{BPQ} \subset \text{BPP}$

Vraag 4. Neem het volgende beslissingsprobleem:

OPSPANNENDE BOOM MET MINIMUM AANTAL BLADEREN

Instantie: een ongerichte graaf G en een positieve integer K .

Vraag: is er een opspannende boom voor G met hoogstens K bladeren?

Voor de complexiteit van dit probleem geldt:

- a. het is een **P**-probleem, immers het vinden van een opspannende boom van een graaf is een polynomiaal probleem.
- b. het is een **NP hard**-probleem, immers HAMP is een restrictie van dit probleem.
- c. het is een **co-NP** probleem, immers nee-instanties van dit probleem zijn efficiënt verifieerbaar.
- d. HAMP is op te lossen in polynomiale tijd als een orakel-aanroep naar dit probleem wordt gedaan met $K=2$.

Vraag 5. Welke van onderstaande beweringen is/zijn correct?

- a. Elk probleem in $\text{NP} \cup \text{co-NP}$ is een P^{NP} probleem.
- b. Sterk **NP**-volledige problemen hebben geen approximatie-algoritme met een approximatie-ratio $1+c$ voor een constante $c > 0$.
- c. Elk **PSPACE-compleet** probleem is een **NP-hard** probleem.
- d. Elk **P**-probleem A is reduceerbaar naar VERTEX COVER.

Vraag 6. Als $P = NP$ dan geldt in ieder geval dat:

- a. $NP \subseteq P$.
- b. $coNP = NP$.
- c. $NP = PSPACE$.
- d. $APX = NPO$.

Vraag 7. 4SAT is het probleem om, gegeven een verzameling U van n propositiesymbolen en een verzameling C van m clauses bestaande uit exact 4 literals over U , te bepalen of C vervulbaar is. Neem het volgende *probabilistische algoritme*:

Gegeven een instantie (U, C) van 4SAT, retourneer een random toekenning τ waarbij voor iedere $u \in U$ geldt: $\Pr(\tau(u) = 1) = \Pr(\tau(u) = 0) = 0.5$

Er geldt nu dat het verwachte aantal clauses dat τ waarmaakt gelijk is aan:

- a. $m \times (1 - 0.5^4)$
- b. $0.5^4 \times m$
- c. $(1 - 0.5^m) \times n$
- d. $(1 - (1 - 0.5^n)) \times m$

Vraag 8. Het is bekend dat MINCOVER een 2-approximatie algoritme M heeft. Laat voor iedere instantie x van MINCOVER, $OPT(x)$ de grootte van de minimum cover aanduiden en $M(x)$ de grootte van de door M gevonden cover voor x . Dan geldt voor elke instantie x van MINCOVER:

- a. $OPT(x) \leq 2$
- b. $OPT(x) / M(x) \leq 2$
- c. $M(x) / OPT(x) \leq 2$
- d. $M(x) - OPT(x) \leq 2$

Vraag 9. Stel dat er een NP-hard probleem X bestaat, dat tevens een P-probleem blijkt te zijn. Welke van de volgende uitspraken is of zijn in dat geval waar?

- a. $NP = coNP$
- b. $NP = P$
- c. $P = NP$
- d. $coNP \cap NP = P$

Vraag 10. Welke van de volgende uitspraken is juist?

- a. $SPACE(n) = NSPACE(n)$.
- b. $\bigcup_{k \geq 0} SPACE(n^k) = \bigcup_{k \geq 0} NSPACE(n^k)$.
- c. $TIME(n^k) \subseteq SPACE(n^k)$.
- d. $NP \subseteq PSPACE$.

OPEN VRAGEN I (30 PUNTEN)

Onderstaande vragen dient u kort te beantwoorden.
Een antwoord dient in hoogstens 5 regels gegeven te worden.

- Vraag 11. Gegeven is een **PSPACE**-volledig probleem X .
Waarom geldt voor ieder **NP**-probleem A , dat $A \leq X$?
- Vraag 12. Laat zien dat $\text{NP}^{\text{NP}} = \text{NP}^{\text{coNP}}$.
- Vraag 13. Toon aan dat $\text{PSPACE}^{\text{PSPACE}} = \text{PSPACE}$.
- Vraag 14. Van een gegeven probleem A is bekend dat $A^c \in \text{coNPC}$.
Laat zien, dat hieruit volgt $A \in \text{NPC}$.
- Vraag 15. Welk voorbeeld kun je noemen om aan te geven dat met quantum computing sommige **NP**-problemen in polynomiale tijd oplosbaar zijn, waarvoor tot nu toe geen (klassiek) polynomiaal algoritme gevonden is?

OPEN VRAGEN II (20 PUNTEN)

Beantwoord onderstaande vragen zo kort mogelijk. Geef geen irrelevante informatie, dit kan tot puntenaftrek aanleiding geven.

Vraag 16. (5 punten)

Dr. I.N. Competent formuleert de volgende stelling met bijbehorend bewijs:

Stelling: $\text{NP} \cap \text{coNP} = \text{P}$

Bewijs: Stel $L \in \text{NP} \cap \text{coNP}$.

Omdat $L \in \text{NP}$, bestaat er een certificaat y van lengte $O(|x|^k)$ voor elke $x \in L$ dat gebruikt kan worden om in polynomiale tijd te verifiëren dat $x \in L$.

Omdat $L \in \text{coNP}$, bestaat er een certificaat z van lengte $O(|x|^k)$ voor elke $x \notin L$ dat gebruikt kan worden om in polynomiale tijd te verifiëren dat $x \notin L$.

Maar dan volgt dat L in polynomial tijd beslist kan worden:

Om te beslissen of $x \in L$ of $x \notin L$, starten we een parallel verificatieproces naar de certificaten y and z . We stoppen zodra het eerste certificaat is geverifieerd. Als het eerste certificaat y is, accepteren we x , anders verwerpen we x .

Dit proces zal zeker stoppen: de maximale tijd benodigd is het maximum van de tijd benodigd voor de verificatie van y of z . Derhalve is dit een polynomiaal algoritme om L te beslissen.

1. Vind de fout(en) in dit bewijs (≤ 5 regels)

Vraag 17. (15 punten)

Een firma heeft twee vrachtwagens en bezorgt elke dag een aantal pakketten bij klanten. Iedere vrachtwagen start vanuit de vestiging van de firma en moet aan het eind van de dag weer terug zijn op dezelfde locatie. Beide vrachtwagens hebben een beperkte actieradius. Vandaar dat de firma graag een oplossing wil vinden voor het volgende beslissingsprobleem:

TWOCYCLEDRIE

Instantie: een verzameling L van locaties, voor iedere paar $x, y \in L$ een afstand $d(x, y) \in \mathbb{Z}^+$, een startlocatie $s \in L$ en een positieve integer K ;

Vraag: zijn er twee simple cycles S_1 en S_2 van locaties in L te vinden die beide s bevatten zodat voor iedere locatie $x \in L$ geldt dat x voorkomt in S_1 of in S_2 en iedere cycle S_i een lengte heeft die hoogstens gelijk is aan K ?

1. Bewijs dat dit probleem een **NP** probleem is (≤ 5 regels).
2. Bewijs dat dit probleem een **NP-hard** probleem is door een reductie van TSP naar dit probleem te geven (≤ 15 regels).

EINDE TENTAMEN