Beknopte antwoorden opgaven werkcollege 9

Opgave 49

We controleren eerst of elke knoop even graad heeft en of de graaf samenhangend is. Dat is het geval. We kunnen nu een Eulerkring construeren. Bijvoorbeeld:

1, 2, 13, 1
Knoop 1 heeft geen onbewandelde takken meer, dus we gaan over onbewandelde takken lopen vanaf 2 totdat we weer bij 2 terugkomen:
1, **2, 3, 4, 5, 6, 12, 9, 11, 3, 10, 4, 9, 2**, 13, 1
Knoop 2 heeft nog steeds onbewandelde takken, dus wandelen we weer vanaf 2 tot we opnieuw 2 tegenkomen:
1, **2, 8, 7, 6, 9, 7, 2**, 3, 4, 5, 6, 12, 9, 11, 3, 10, 4, 9, 2, 13, 1
Er zijn geen onbewandelde takken meer, dus klaar.

Opgave 51

De vraag is niet helemaal goed gesteld; een Eulerkring is immers ook een Eulerpad. We herformuleren de opgave dus als volgt: $\mathcal{G}$ bevat een Eulerpad d.e.s.d.a. er precies twee *óf nul* knopen zijn met oneven graad. In dat laatste geval is er een Eulerkring, maar dat is ook een Eulerpad.

Merk op dat het aantal knopen met oneven graad altijd even is (zie het college FoCS uit jaar 1).

- als nul knopen met oneven graad: er bestaat een Eulerkring en dus een Eulerpad.
- als twee knopen met oneven graad: verbind deze twee, maak een Eulerkring (alle knopen hebben nu even graad) en verwijder dan de tak weer (zowel uit de graaf als uit de geconstrueerde kring). Je houdt een Eulerpad over.
- dus als er nul of twee knopen met oneven graad zijn, bestaat er een Eulerpad
- als er meer dan twee knopen met oneven graad zijn: als er een Eulerpad bestaat, dan geldt voor een knoop met oneven graad dat het pad er ofwel moet beginnen of eindigen, anders zit het er uiteindelijk vast (je kan er één keer minder vertrekken dan aankomen). Een Eulerpad kan maar bij één knoop beginnen en bij één knoop eindigen, dus je kan niet meer dan twee knopen hebben met oneven graad als je een Eulerpad wilt maken.

Opgave 52

a. Bijvoorbeeld: 1, 2, 3, 11, 9, 2, 7, 6, 5, 4, 10, 3, 4, 9, 6, 12, 9, 7, 8, 2, 13, 1

Opgave 55

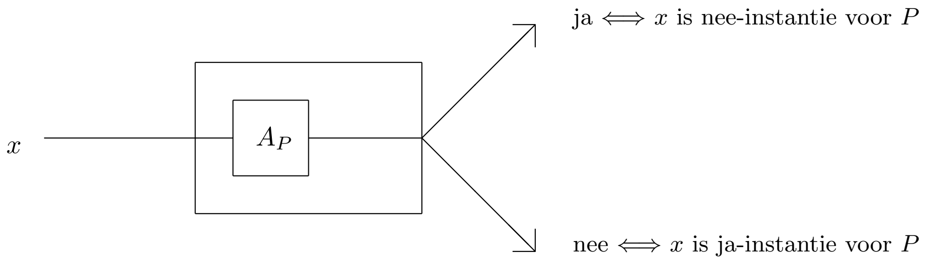
a. Laat de aantallen operaties van A en B resp. $a(n)$ en $b(n)$ zijn, met n de grootte van de invoer. Dan doet C $c(n) = a(n) + b(n)$ operaties voor deze invoer.

Dit is hooguit $2a(n)$ (als $a(n) \geq b(n)$) of $2b(n)$ (als $b(n) \geq a(n)$). Aangezien $a(n)$ en $b(n)$ polynomiaal zijn, is $c(n)$ dat ook.

b. A en B zijn polynomiaal begrensd, dus de complexiteit van A is in $O(n^k)$ en de complexiteit van B is in $O(n^\ell)$ voor constanten k, ℓ. De grootte van de uitvoer van A is dan dus ook in $O(n^k)$. Dan geldt dat $B \circ A \in O((n^k)^\ell) = O(n^{k\ell})$. k en ℓ zijn constanten, dus $B \circ A \in O(n^m)$ voor een of andere constante m, dus $B \circ A$ is polynomiaal begrensd.

Opgave 58

a. - Stel dat $P \in \mathcal{P}$. Te bewijzen is dat $\overline{P} \in \mathcal{P}$. Merk op (zie de opgave) dat de invoerverzameling van P en $\overline{P}$ hetzelfde is, alleen zijn ja-instanties voor P nee-instanties van $\overline{P}$ en omgekeerd. Omdat $P \in \mathcal{P}$ bestaat er een algoritme A_P dat voor elke invoer/instantie van P in polynomiale tijd een ja- of nee-antwoord geeft. We maken nu een nieuw algoritme dat $\overline{P}$ oplost door de invoer/instantie eerst aan A_P te geven, en dan vervolgens precies het tegenovergestelde te antwoorden. Dat wil zeggen, als A_P 'ja' antwoordt, geeft het nieuwe algoritme 'nee' terug en omgekeerd. Dit algoritme geeft voor ja instanties van P het antwoord 'nee' en voor nee-instanties van P het antwoord 'ja', dus is een algoritme voor $\overline{P}$. Het is uiteraard ook polynomiaal, omdat A_P dat is. Dus $\overline{P} \in \mathcal{P}$.



- Stel dat $\overline{P} \in \mathcal{P}$. Omdat $\overline{\overline{P}} = P$ kunnen we dezelfde bewijsvoering gebruiken om aan te tonen dat $P \in \mathcal{P}$.

- Samen geeft dat: $P \in \mathcal{P} \Leftrightarrow \overline{P} \in \mathcal{P}$.

c. Te bewijzen is dat $\mathcal{P} = \text{co-}\mathcal{P} = \{P : \overline{P} \in \mathcal{P}\}$. Dit bewijs volgt direct uit opgave a. Immers voor elke P geldt: $P \in \mathcal{P} \Leftrightarrow \overline{P} \in \mathcal{P} \Leftrightarrow P \in \text{co-}\mathcal{P}$.

Opgave 2 tentamen 4/6/19

Zie uitwerking website