Tentamen Geometrische Algoritmen, 8 mei 2001, 14–17 uur

Naam: .... **Collegekaartnummer:** ....

Voor dit tentamen zijn 9.5 punten te halen (0.5 punt is kado). Lees elke opgave zorgvuldig door voordat je 'm beantwoordt; zorg dat je begrijpt wat er gevraagd wordt. Controleer achteraf of je inderdaad de vraag hebt antwoord. Maak eerst de opgaven die je makkelijk vindt, en daarna pas de moeilijkere. Het tentamen is gesloten boek.

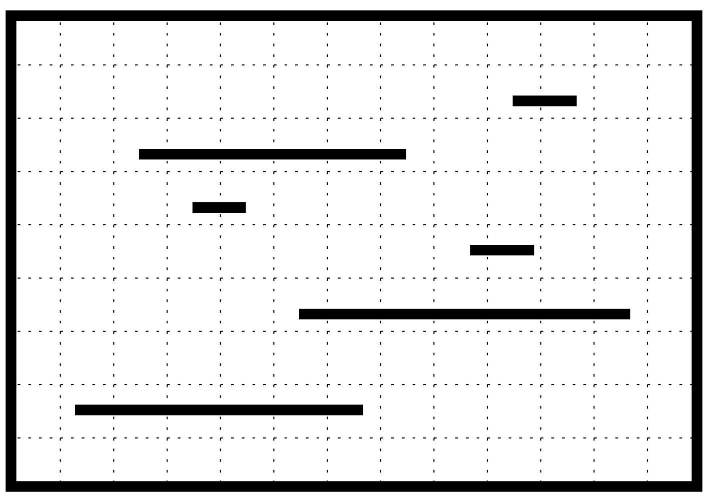
1. **(a.)** (1 punt) Beschrijf precies hoe een intervalboom voor een verzameling van (1-dimensionale) intervallen is opgebouwd.

   (b.) (0.5 punt) Geef het zoekalgoritme om alle intervallen in de intervalboom te rapporteren die een gegeven zoekwaarde x bevatten.

2. (1 punt) Gegeven een convex polygoon P met alle vertices in een array $A[0,\dots,n-1]$. De meest linkse vertex van P zit in $A[0]$ (beide coördinaten zitten daar dus) en de vertices zijn tegen de klok in opgeslagen in A.

   Geef een algoritme dat in $O(\log n)$ tijd bepaalt of een gegeven punt q in het polygoon P ligt of niet.

3. Gegeven een verzameling H met n horizontale lijnstukken in het platte vlak. De lijnstukken zijn niet-snijdend, en geen twee eindpunten van verschillende lijnstukken hebben dezelfde x- of y-coördinaat.

   (a.) (1 punt) Teken de trapezoïdale decompositie (vertikale decompositie) binnen de getekende bounding box van de 6 lijnstukken in de figuur hierboven. De gestippelde lijnen zijn alleen voor oriëntatie en horen niet tot de lijnstukken waarvoor de trapezoïdale decompositie getekend moet worden.

   (b.) (2 punten) Geef een sweep algoritme dat alle lijnstukken oplevert die (naast de delen van H) in de trapezoïdale decompositie zitten. Beschrijf de status en statusstructuur, en geef de events, de eventlijst en de eventafhandeling. Het algoritme moet in $O(n \log n)$ tijd werken, maar dit hoeft niet bewezen te worden.

4. Beschouw het algoritme voor lineair programmeren in 2 dimensies, waarvan een gewone incrementele en een gerandomiseerde incrementele versie bestaat. We beschouwen de versies uit het boek (hieronder staat die voor gewone incrementele LP).

Algorithm 2DBOUNDEDLP$(H, \vec{c}, m_1, m_2)$

Input: A linear program $(H \cup \{m_1, m_2\}, \vec{c})$, where H is a set of n half-planes, $\vec{c} \in R^2$, and m_1, m_2 are two half-planes that bound the solution.
Output: If $(H \cup \{m_1, m_2\}, \vec{c})$ is infeasible, then this fact is reported. Otherwise, the lexicographically smallest point p that maximizes $f_{\vec{c}}(p)$ is reported.

Let v_0 be the point that maximizes $f_{\vec{c}}(\{m_1, m_2\})$.
Let $h_1, \ldots, h_n$ be the half-planes of H.
for $i \leftarrow 1$ **to** n
do if $v_{i-1} \in h_i$
  then $v_i \leftarrow v_{i-1}$
  else $v_i \leftarrow$ the point p on ℓ_i that maximizes $f_{\vec{c}}(p)$, subject to the constraints in H_{i-1}.
    if p does not exist
    then Report that the linear program is infeasible and quit.
return v_n

Neem aan: $m_1 := (x + y \geq 0)$ en $m_2 := (y - x \geq 0)$ in de volgende onderdelen. Neem ook aan dat de objective function $f_{\vec{c}}(..)$ gemaximaliseerd wordt bij het laagste punt, dus minimale y-coördinaat.

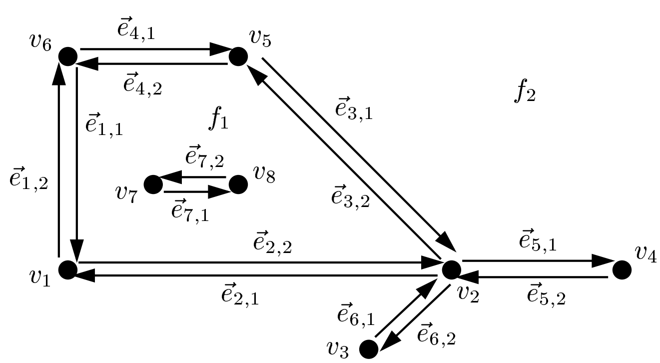
(a.) (0.5 punt) Wat is de worst-case tijdgrens als het gewone incrementele algoritme wordt toegepast op de verzameling half-vlakken (in onbekende volgorde)

$$H = \{x + y \geq k \mid 1 \leq k \leq n/2\} \cup \{y - x \geq k \mid 1 \leq k \leq n/2\}\ .$$

(b.) (0.5 punt) Wat is de verwachte tijdgrens van het gerandomiseerde algoritme als het op de verzameling H met n half-vlakken wordt toegepast?

(c.) (0.5 punt) Bestaat er een verzameling H met n half-vlakken waarvoor het LP feasible is, en waarvoor het gewone incrementele algoritme in het worst-case geval toch slechts $O(n)$ tijd nodig heeft? Beargumenteer het antwoord of geef een voorbeeld.

5. In de figuur hierna staan de vertices, half-edges en faces van een subdivisie in doubly-connected edge list getekend.



(a.) (0.5 punt) Geef aan naar welke half-edges, vertices en faces de half-edge record van $\vec{e}_{2,1}$ verwijzingen (pointers) heeft in de doubly-connected edge list structuur.

(b.) (0.5 punt) Geef aan naar welke half-edges, vertices en faces de face record van f_1 verwijzingen (pointers) heeft. Er zijn meerdere mogelijke antwoorden; geef een antwoord dat bij een mogelijke DCEL voor de getekende subdivisie kan horen.

6. (1.5 punten) Een polygoon P heet *stervormig* als er een punt q in P ligt (eventueel op de rand van P) zodanig dat voor elk ander punt r in P (eventueel op de rand van P), het lijnstuk $\overline{qr}$ volledig in P ligt (eventueel deels met de rand samenvalt), en dus niet buiten het polygoon P komt.

Geef een zo efficiënt mogelijk algoritme om te bepalen of een polygoon P met n vertices stervormig is. Hoe snel is je algoritme?