

Question 1: Alle-til-alle vha. Dijkstra

Hvordan kan du løse alle til alle korteste vei-problemet i en rettet graf med ikke-negative kantvektter, ved hjelp av Dijkstras algoritme?

- ☒ Kjør algoritmen en gang fra hver node, totalt $\Theta(V)$ ganger.
- ☐ Ingen av de andre alternativene er korrekte.
- ☐ Kjør algoritmen mellom alle par med noder, totalt $\Theta(V^2)$ ganger.
- ☐ Kjør algoritmen en gang.

Question 2: Alle-til-alle vha. SSSP

Hvordan kan du raskest finne korteste vei mellom alle par med noder i en rettet graf uten negative sykler, ved å kjøre en av algoritmene nedenfor fra hver node? Hva blir kjøretiden?

- ☐ Dijkstra m/min-heap, $O(VE \lg V)$
- ☐ Bellman-Ford, $O(V^3)$
- ☐ Dijkstra m/min-heap $O(V^2 \lg V + VE)$
- ☒ Bellman-Ford, $O(V^2 E)$

Question 3: Forgjengermatriser

π_{ij} i en forgjengermatrise, forteller oss

- ☐ Hvor man må gå for å ta korteste vei fra i til j
- ☐ Hvor man må gå for å ta korteste vei fra j til i
- ☒ Hvor man kom fra, på korteste vei fra i til j
- ☐ Hvor man kom fra, på korteste vei fra j til i

Question 4: Forgjengermatriser

$\pi_{ij} = nil$ betyr at

- ☐ Det er aldri mulig å komme seg fra i til j
- ☐ Enten er $i = j$ eller så er det ingen sti fra j til i
- ☒ Enten er $i = j$ eller så er det ingen sti fra i til j
- ☐ Ingen av de andre alternativene er korrekte.

Question 5: Floyd-Warshall

Hva er kjøretiden til Floyd-Warshall?

- ☐ $O(V^2 \lg V)$
- ☒ $O(V^3)$
- ☐ $O(V^4)$
- ☐ $O(VE)$

Question 6: Floyd-Warshall

Floyd-Warshall bruker vanligvis nabolister for å representere grafen

- ☐ Sant.
- ☒ Usant.

Question 7: Floyd-Warshall

Implementasjonen av Floyd-Warshall i kapittel 25.2 i Cormen bruker unødvendig mye plass.

I praksis bruker vi en versjon som bruker mindre plass. Hvor mye plass bruker denne implementasjonen?

- ☒ $\Theta(V^2)$
- ☐ $\Theta(E + V)$
- ☐ $\Theta(V^3)$
- ☐ $\Theta(E)$

Question 8: Floyd-Warshall

Ta stilling til følgende utsagn:

1. Etter at Floyd-Warshall har kjørt, kan diagonalen avstandsmatrisen D (dvs. $d_{1,1}$, $d_{2,2}$ osv.) inneholde positive tall.
2. Etter at Floyd-Warshall har kjørt, kan diagonalen avstandsmatrisen D (dvs. $d_{1,1}$, $d_{2,2}$ osv.) inneholde negative tall.

- ☒ Begge utsagnene er sanne.
- ☐ Begge utsagnene er usanne.
- ☐ Kun utsagn 2 er sant.
- ☐ Kun utsagn 1 er sant.

Question 9: Transitive-closure

Oppgave fra en tidligere eksamen:

I Transitive-Closure brukes den binære variabelen $t_{ij}^{(k)}$ til å indikere om det går en sti fra i til j hvis alle noder på veien mellom dem *må* ligge i mengden $1, 2, \dots, k$. For eksempel er $t_{ij}^{(0)} = 1$ hvis og bare hvis $(i, j) \in E$. Hva er uttrykket for $t_{ij}^{(k)}$, når $k > 0$?

- ☐ $t_{ij}^{(k)} = t_{ij}^{(k-1)} \wedge (t_{ik}^{(k-1)} \vee t_{kj}^{(k-1)})$
- ☐ $t_{ij}^{(k)} = t_{ij}^{(k-1)} \vee (t_{jk}^{(k-1)} \wedge t_{ki}^{(k-1)})$
- ☐ $t_{ij}^{(k)} = t_{ij}^{(k-1)} \wedge (t_{jk}^{(k-1)} \vee t_{ki}^{(k-1)})$
- ☒ $t_{ij}^{(k)} = t_{ij}^{(k-1)} \vee (t_{ik}^{(k-1)} \wedge t_{kj}^{(k-1)})$

Question 10: Transitive-closure

Dersom $t_{ij}^{(k)} = 0$ betyr det at

- ☐ Det eksisterer en sti fra i til j med lengde nøyaktig lik k .
- ☐ Det eksisterer en sti fra i til j med lengde større eller lik k .
- ☐ Det ikke eksisterer en sti fra i til j med lengde større eller lik k .
- ☒ Ingen av de andre alternativene.
- ☐ Det eksisterer en sti fra i til j med lengde mindre eller lik k .
- ☐ Det ikke eksisterer en sti fra i til j med lengde mindre eller lik k .

Question 11: Johnsons algoritme

Johnsons bruker andre algoritmer som subrutiner. Hvilke?

- ☐ Dijkstra og BFS
- ☐ Bellman-Ford og Floyd-Warshall
- ☒ Dijkstra og Bellman-Ford
- ☐ BFS og Floyd-Warshall

Question 12: Johnsons algoritme

Anta at vi bruker en binær min-heap. Da har Johnsons algoritme har kjøretid

- ☐ $O(V^2 \lg V + VE)$
- ☐ $O(V^2 \lg V + V^3)$
- ☐ $O(E \lg V + VE)$
- ☒ $O(VE \lg V)$

Question 13: Johnsons algoritme

Johnsons algoritme finner korteste vei i grafer med negative sykler.

- ☒ Usant.
- ☐ Sant.

Question 14: Johnsons algoritme

Hvilken teknikk er det som gjør Johnsons algoritme spesiell?

- ☐ Grådighet
- ☒ Revekting av kantvekter
- ☐ Bruk av nabolister
- ☐ Relaksering av kantvekter

Question 15: Johnsons algoritme

Hva blir kjøretiden til Johnsons algoritme i en rettet graf der alle par med noder har en kant hver vei mellom seg (en komplett digraf), dersom vi antar at vi bruker en Fibonacci-heap?

- ☐ $O(VE)$
- ☐ $O(V^4)$
- ☒ $O(V^3)$
- ☐ $O(V^2 \lg V + VE)$