Øving 11, teori: Alle til alle

(/inginious/course/TDT4120/10p)

(/inginious/course/TDT4120/11p)

Your answer passed the tests! Your score is 100.0%. [Submission #5c0052967f80cc0beb1f1f7c] ×

Question 1: Alle-til-alle vha. Dijkstra

×

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

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

Kommentar til løsning:
Vi har en en graf uten negative sykler, men oppgaven sier ingenting om negative kanter.

Dermed kan det være negative kanter, og Dijkstra vil ikke fungere.

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

Question 3: Forgjengermatriser

π_{ij} i en forgjengermatrise, forteller oss

Kommentar til løsning:
π_{ij} er *forgjengeren* til j, når man går fra i.

- ( ) Hvor man kom fra, på korteste vei fra j til i
- ( ) 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

Question 4: Forgjengermatriser

$\pi_{ij} = nil$ betyr at

- ( ) Ingen av de andre alternativene er korrekte.
- ( ) 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

Question 5: Floyd-Warshall

Hva er kjøretiden til Floyd-Warshall?

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

Question 6: Floyd-Warshall

Floyd-Warshall bruker vanligvis nabolister for å representere grafen.

Kommentar til løsning:
Floyd-Warshall bruker nabomatriser.

- (x) Usant.
- ( ) Sant.

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?

Kommentar til løsning:
Vi trenger ikke lage en ny matrise for hver k. Se pseudokode i oppgave 25.2-4.

- ○ $\Theta(E+V)$
- ● $\Theta(V^2)$
- ○ $\Theta(E)$
- ○ $\Theta(V^3)$

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.

Kommentar til løsning:
Dersom vi har negative sykler få diagonalen negative verdier i nodene som er en del av en negativ sykel.

- ○ 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 $1, 2, \ldots, k$. For eksempel er $t_{ij}^{(0)} = 1$ hvis og bare hvis $(i, j) \in E$. Hva er utrykket for $t_{ij}^{(k)}$, når $k > 0$?

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

Question 10: Transitive-closure

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

Kommentar til løsning:
Dersom $t_{ij}^{(k)} = 0$ vet vi at det ikke finnes en kant fra i til j, og at det ikke går en sti mellom nodene gjennom nodene $1, \ldots, k$.

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

Question 11: Johnsons algoritme

Johnsons bruker andre algoritmer som subrutiner.

Hvilke?

- [ ] Dijkstra og BFS
- [ ] BFS og Floyd-Warshall
- [ ] Bellman-Ford og Floyd-Warshall
- [x] Dijkstra og Bellman-Ford

Question 12: Johnsons algoritme

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

Kommentar til løsning:
Definisjon fra Cormen.

- [x] $O(VE \lg V)$
- [ ] $O(V^2 \lg V + VE)$
- [ ] $O(E \lg V + VE)$
- [ ] $O(V^2 \lg V + V^3)$

Question 13: Johnsons algoritme

Johnsons algoritme finner korteste vei i grafer med negative sykler.

- [ ] Sant.
- [x] Usant.

Question 14: Johnsons algoritme

Hvilken teknikk er det som gjør Johnsons algoritme spesiell?

- [ ] Grådighet
- [x] 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?

Kommentar til løsning:
Fra oppgaveteksten vet vi at vi har $E = O(V^2)$ kanter. Kjøretiden for Johnson's algoritme med Fibonacci-heap blir da $O(V^2 \lg V + VE) = O(V^2 \lg V + V^3) = O(V^3)$.

- [x] $O(V^3)$
- [ ] $O(VE)$
- [ ] $O(V^4)$
- [ ] $O(V^2 \lg V + VE)$

Submit

>_