

Score: 0.250 (=100.0%)

Id: 45547

Što je razlog za računalnu složenost Kruskalovog algoritma?

**a** sortiranje bridova ulaznog grafa silazno po oznaci ili rednom broju

**b** sortiranje bridova ulaznog grafa uzlazno po oznaci ili rednom broju

**c** sortiranje bridova ulaznog grafa uzlazno po težinama

**d** sortiranje bridova ulaznog grafa silazno po težinama

Score: 0.250 (=100.0%)

Id: 45539

Računalna složenost Dijkstrinog algoritma (MST) je:

**a**  $O(V * E)$

**b**  $O(V + E)$

**c**  $O(V)$

**d**  $O(E)$

Score: 0.250 (=100.0%)

Id: 45541

Najvažniji nedostatak Kruskalovog algoritma spram Dijkstrinog algoritma (MST) je:

**a** visoka prostorna složenost zbog potrebe za korištenjem stoga

**b** visoka računalna složenost zbog potrebe za sortiranjem bridova

**c** visoka vremenska složenost zbog potrebe za višestrukim obilaskom grafa

**d** visoka računalna složenost zbog potrebe za sortiranjem vrhova

Score: 0.250 (=100.0%)

Id: 45543

Hierholzerov algoritam temelji se na...

- ☒ a činjenici da je Eulerov graf povezan pa je moguće upotrijebiti stog za izbjegavanje slijepog pretraživanja svih vrhova u potrazi za novim bridom
- ☐ b činjenici da je Eulerov graf povezan pa je moguće upotrijebiti stog za pomoći pri izbjegavanju slijepog pretraživanja nekih vrhova u potrazi za novim bridom
- ☐ c činjenici da je Eulerov graf nepovezan pa je moguće upotrijebiti stog za pomoći pri izbjegavanju slijepog pretraživanja nekih vrhova u potrazi za novim bridom
- ☐ d činjenici da je Eulerov graf ponekad povezan pa je moguće upotrijebiti stog za izbjegavanje slijepog pretraživanja svih vrhova u potrazi za novim bridom

Računalna složenost Kruskalovog algoritma je:

- ☐ a  $O(V^n \log_{10} V)$
- ☒ b  $O(V^2 \log_2 V)$
- ☐ c  $O(E^n \log_{10} V)$
- ☐ d  $O(1)$

Označite najvažniji nedostatak najjednostavnijeg načina za detekciju ciklusa u grafu korištenjem DFS (*Depth-first search*) strategije:

- ☐ a Velika računalna složenost postupka obilaska
- ☐ b Zauzeće radne memorije potrebno za implementaciju postupka obilaska
- ☐ c Postupak nije univerzalno primjenjiv na sve grafove
- ☒ d Nikad ne možemo biti sigurni da li smo detektirali ciklus ako u ponovnom spuštanjem kroz graf naiđemo na prethodno označeni vrh

Označite korake najjednostavnijeg načina za detekciju ciklusa u grafu korištenjem DFS (*Depth-First Search*) ili BFS (*Breadth-First Search*) strategija za obilazak grafa:

**a** označimo svaki vrh koji obiđemo

**b** ako u obilasku naiđemo na već označeni vrh kojega nema na stogu, tada smo detektirali ciklus u grafu

**c** označimo svaki vrh koji obiđemo i stavimo ga na stog

**d** ako u obilasku naiđemo na već označeni vrh, tada smo detektirali ciklus u grafu

Dijkstrin algoritam (MST) razlikuje se od Kruskalovog algoritma jer:

**a** U trenutku kada detektiramo ciklus, iz ciklusa uklanjamo brid najveće težine

**b** Nema sortiranja

**c** U trenutku kada detektiramo ciklus, iz ciklusa uklanjamo brid najmanje težine

**d** Ima sortiranja

Score: 0.250 (=100.0%)

Id: 45550

Ukupna računalna složenost Primovog algoritma u implementaciji s gomilom je:

**a**  $O(E + V \log_2 V)$

**b**  $O(V^3)$

**c**  $O(V + E \log_2 E)$

**d**  $O(V^2)$

Score: 0.250 (=100.0%)

Id: 45537

Označite postupke koji se koriste za detekciju ciklusa u grafu korištenjem stoga:

**a** označavamo vrhove obijeđenim i koristimo stog  $S$  s kojim pratimo trenutnu putanju

**b** ulaskom u rekurzivni poziv mičeemo vrh sa stoga  $S$ , a izlaskom stavljamo ga na stog

**c** označavamo vrhove obijeđenim i koristimo stog  $S$  s kojim pratimo prethodnu putanju

**d** ulaskom u rekurzivni poziv stavljamo vrh na stog  $S$ , a izlaskom ga mičeemo sa stoga

Ukupna računalna složenost Primovog algoritma u sekvencijalnoj implementaciji je:

**a**  $O(V + E \log_2 E)$

**b**  $O(V^2)$

**c**  $O(E + V \log_2 V)$

**d**  $O(V^4)$

Računalna složenost Hierholzerovog algoritma je:

**a**  $O(V * E)$

**b**  $O(V)$

**c**  $O(V + E)$

**d**  $O(E)$

Kruskalov algoritam može se primijeniti:

**a** isključivo za usmjerene grafove

**b** za grafove s barem jednim ciklusom

**c** za usmjerene i neusmjerene grafove

**d** isključivo za neusmjerene grafove