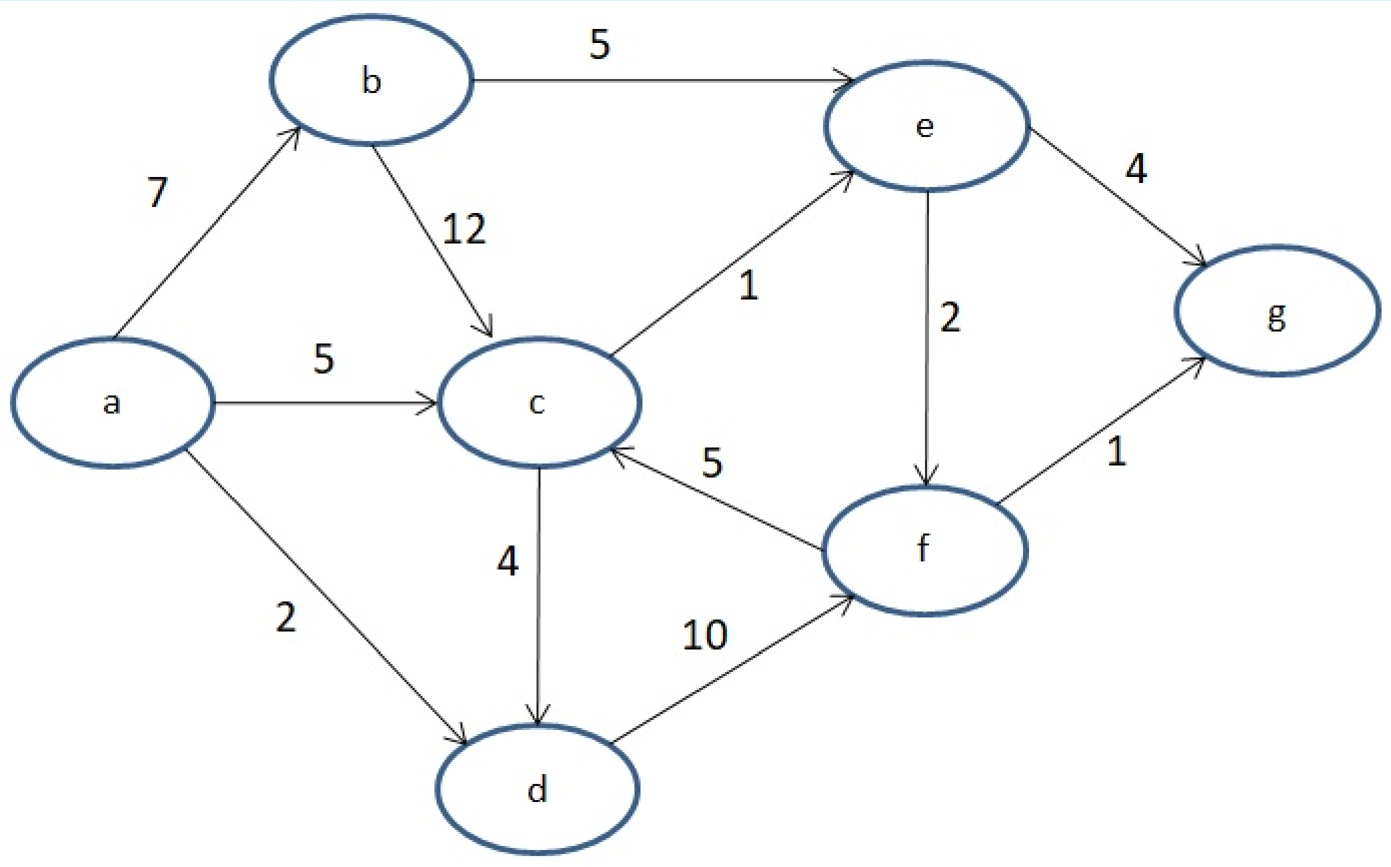
Začeto dne	četrtek, 16. januar 2020, 09:53
Stanje	Zaključeno
Dokončano dne	ponedeljek, 20. januar 2020, 00:00
Porabljeni čas	3 dni 14 ure
Točke	6,25/8,00
Ocena	**7,81** od možne ocene 10,00 (**78**%)

Vprašanje **1**
Pravilno
Ocena 1,00 od 1,00

Na sliki je usmerjeni graf.



Z Dijkstrinim algoritmom želimo zgraditi vpeto drevo najkrajših povezav iz vozlišča A. V kakšnem vrstnem redu bodo dodana vozlišča v vpeto drevo (ki že vsebuje vozlišče A, kot koren vpetega drevesa)?

Prvo dodano vozlišče	d ✔
Drugo dodano vozlišče	c ✔
Tretje dodano vozlišče	e ✔
Četrto dodano vozlišče	b ✔
Peto dodano vozlišče	f ✔
Šesto dodano vozlišče	g ✔

Vaš odgovor je pravilen.

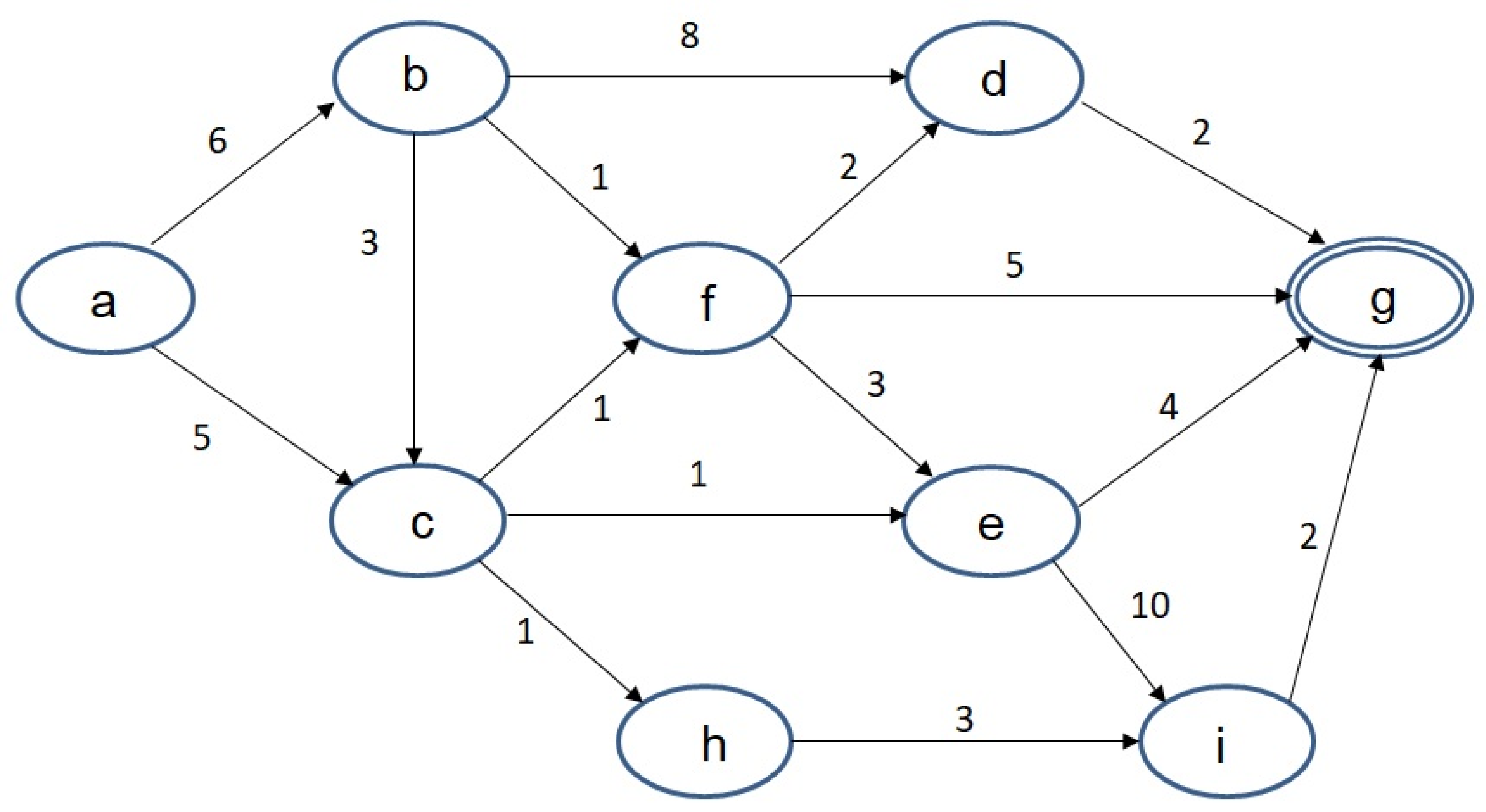
Pravilen odgovor je: Prvo dodano vozlišče → d, Drugo dodano vozlišče → c, Tretje dodano vozlišče → e, Četrto dodano vozlišče → b, Peto dodano vozlišče → f, Šesto dodano vozlišče → g

Vprašanje **2**

Pravilno

Ocena 1,00 od 1,00

Na sliki je prikazan mrežni diagram: a je začetno vozlišče, g pa zaključno.



Z algoritmom za analizo kritične poti po principu dinamičnega programiranja poiščemo pot med vozliščema a in g.

Naslednik vozlišča b na kritični poti je	c ✔
Naslednik vozlišča d na kritični poti je	vozlišče ni na kritični poti ✔
Naslednik vozlišča i na kritični poti je	g ✔
Naslednik vozlišča f na kritični poti je	e ✔
Naslednik vozlišča e na kritični poti je	i ✔
Naslednik vozlišča h na kritični poti je	vozlišče ni na kritični poti ✔
Naslednik vozlišča a na kritični poti je	b ✔
Naslednik vozlišča c na kritični poti je	f ✔

Vaš odgovor je pravilen.

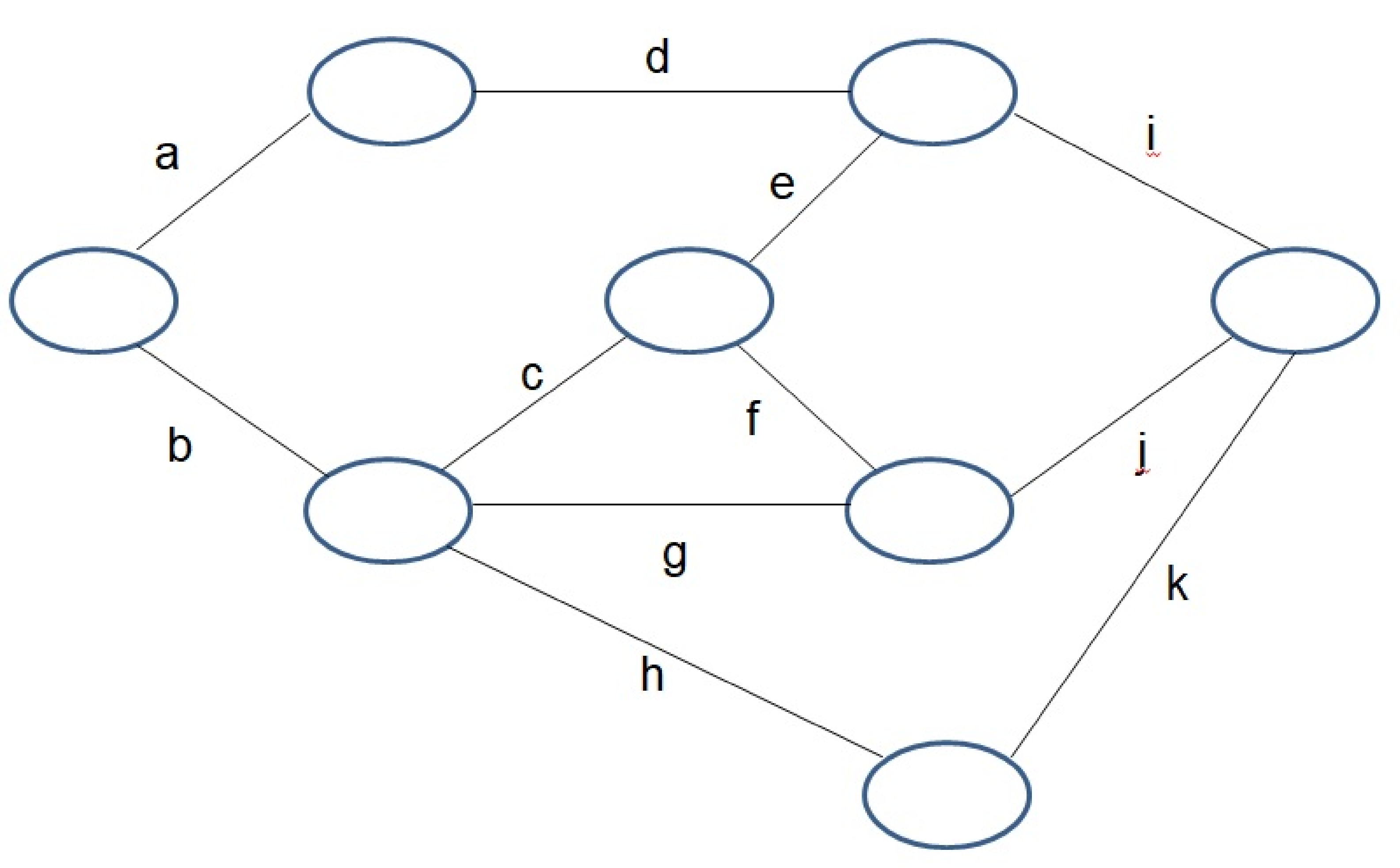
Pravilen odgovor je: Naslednik vozlišča b na kritični poti je → c, Naslednik vozlišča d na kritični poti je → vozlišče ni na kritični poti, Naslednik vozlišča i na kritični poti je → g, Naslednik vozlišča f na kritični poti je → e, Naslednik vozlišča e na kritični poti je → i, Naslednik vozlišča h na kritični poti je → vozlišče ni na kritični poti, Naslednik vozlišča a na kritični poti je → b, Naslednik vozlišča c na kritični poti je → f

Vprašanje **3**

Pravilno

Ocena 1,00 od 1,00

Na sliki je neusmerjeni graf.



Cene povezav so: a(6), b(6), c(2), d(2), e(6), f(3), g(4), h(5), i(5), j(4), k(2). S Kruskalovim algoritmom želimo zgraditi minimalno vpeto drevo. Če je več povezav enakovrednih, potem predpostavimo, da algoritem izbira po abecedi. V kakšnem vrstnem redu algoritem dodaja povezave v gozd?

Prva dodana povezava	c ✔
Druga dodana povezava	d ✔
Tretja dodana povezava	k ✔
Četrta dodana povezava	f ✔
Peta dodana povezava	j ✔
Šesta dodana povezava	i ✔
Sedma dodana povezava	a ✔

Vaš odgovor je pravilen.

Pravilen odgovor je: Prva dodana povezava → c, Druga dodana povezava → d, Tretja dodana povezava → k, Četrta dodana povezava → f, Peta dodana povezava → j, Šesta dodana povezava → i, Sedma dodana povezava → a

Vprašanje **4**
Pravilno
Ocena 1,00 od 1,00

Nekdo se je lotil iskanja kritične poti v grafu tako, da je trivialno spremenil Dijkstrin algoritem: elemente v kopici je uredil od največjega proti najmanjšemu. Na ta način je v vsaki iteraciji v vpeto drevo dodal vozlišče, ki ima najdaljšo znano pot od začetnega vozlišča. Katera trditev je pravilna?

- (●) opisani algoritem ne bo vedno pravilno deloval, čeprav ga poganjamo na usmerjenih grafih brez ciklov in brez povezav z negativnimi cenami ✔
- ( ) opisani algoritem bo pravilno deloval na vseh usmerjenih grafih brez ciklov in brez povezav z negativnimi cenami
- ( ) opisani algoritem bo pravilno deloval na vseh usmerjenih grafih brez ciklov (ne glede na morebitne negativne cene povezav)
- ( ) opisani algoritem bo pravilno deloval na vseh usmerjenih grafih (ne glede na morebitne cikle in negativne cene povezav)

Vaš odgovor je pravilen.

Pravilen odgovor je: opisani algoritem ne bo vedno pravilno deloval, čeprav ga poganjamo na usmerjenih grafih brez ciklov in brez povezav z negativnimi cenami

Vprašanje **5**
Pravilno
Ocena 1,00 od 1,00

S Kruskalovim algoritmom želimo poiskati MAKSIMALNO vpeto drevo za podani graf. Katere trditve so pravilne?

Izberite enega ali več odgovorov:

- [ ] trivialna sprememba algoritma, s katero bi v vsaki iteraciji izbirali maksimalno namesto minimalne povezave med dvema vpetima drevesoma, ne bo dala pravilnega rezultata
- [ ] cene povezav v grafu bi lahko množili z -1, toda Kruskalov algoritem ne deluje na grafih z negativnimi cenami povezav
- [x] če algoritem spremenimo tako, da v vsaki iteraciji izbere maksimalno povezavo med dvema vpetima drevesoma, dobimo maksimalno vpeto drevo grafa ✔
- [x] če cene vseh povezav v grafu množimo z -1 in poženemo algoritem, dobljena rešitev določa maksimalno vpeto drevo originalnega grafa ✔

Vaš odgovor je pravilen.

Pravilni odgovori so: če cene vseh povezav v grafu množimo z -1 in poženemo algoritem, dobljena rešitev določa maksimalno vpeto drevo originalnega grafa, če algoritem spremenimo tako, da v vsaki iteraciji izbere maksimalno povezavo med dvema vpetima drevesoma, dobimo maksimalno vpeto drevo grafa

Vprašanje **6**
Delno pravilno
Ocena 0,25 od 1,00

Želimo uporabiti Dijkstrin algoritem za gradnjo vpetega drevesa najkrajših poti v grafu. Težava je v tem, da ima naš graf nekatere (ne vse) povezave z negativnimi cenami. Kako lahko spremenimo graf, da bo dobljena rešitev pravilna tudi za začetno situacijo (več možnih odgovorov)?

Izberite enega ali več odgovorov:

- [x] če v grafu obstajajo "negativni cikli", najkrajše poti sploh niso definirane (zaradi negativnih povezav so lahko nekatere poti dolge minus neskončno) ✔
- [ ] cene vseh povezav spremenimo v pozitivne vrednosti tako, da jim prištejemo dovolj veliko konstanto ter nato poženemo algoritem
- [ ] povezave z negativnimi cenami odstranimo iz grafa in nato poženemo algoritem
- [x] grafa ni potrebno spreminjati - dobljeni rezultat bo pravilen, ne glede na negativne cene povezav ✖
- [ ] grafa se v splošnem ne da popraviti tako, da bi Dijkstrin algoritem pravilno deloval
- [ ] cene povezav zamenjamo z njihovimi absolutnimi vrednostmi in nato poženemo algoritem

Vaš odgovor je delno pravilen.

Pravilno ste izbrali 1.
Pravilni odgovori so: grafa se v splošnem ne da popraviti tako, da bi Dijkstrin algoritem pravilno deloval, če v grafu obstajajo "negativni cikli", najkrajše poti sploh niso definirane (zaradi negativnih povezav so lahko nekatere poti dolge minus neskončno)

Vprašanje **7**

Pravilno

Ocena 1,00 od 1,00

Primov algoritem implementiramo tako, da prioritetno vrsto predstavimo z AVL drevesom. Če z n označimo število vozlišč v grafu in z m število povezav, je časovna kompleksnost dobljenega algoritma:

- O(m*log(m))
- O(m*log(n)) ✔
- O(n^2)
- O(m(log(m)+n^2))
- O(m*n)

Vaš odgovor je pravilen.

Pravilen odgovor je: O(m*log(n))

Vprašanje **8**

NEpravilno

Ocena 0,00 od 1,00

Na grafu, ki ima n vozlišč in m povezav, smo pognali Primov algoritem. Pri gradnji minimalnega vpetega drevesa bo algoritem izvršil n-krat operaciji INSERT in DELETEMIN. Velja tudi, da bo algoritem izvršil operacijo DECREASE_KEY največ O(m) krat. Kakšna bo časovna zahtevnost algoritma v odvisnosti od števila vozlišč n in števila povezav m, če prioritetno vrsto realiziramo z urejenim seznamom?

- O(n * m)
- O(n^2 + m*n)
- O(n * log(n)) ✖
- O(m * log(m))
- O(m^2)

Vaš odgovor je napačen.

Pravilen odgovor je: O(n^2 + m*n)

◄ Grafi - rešitve kritična pot, Dijkstra, Kruskal (kopiraj)

Skok na...