

## **Beugró kérdések**

1. Mit nevezünk felejtő és nem felejtő tárolónak?
2. Mi az a RAID?
3. Mit nevezünk puffernek?
4. Mi az a lapcsere (lapozás, paging)?
5. Mit nevezünk piszkos puffernek?
6. Milyen fájlszervezási módok vannak?
7. Mit jelent a kupac fájlszervezés?
8. Mit jelent a rendezett fájlszervezés?
9. Mi az a memóriahozzáférés igazítás (word-alignment)?
10. Milyen céljaink vannak a fájlszervezés kiválasztásánál?
11. Mit nevezünk egyenletességi feltételnek a lekérdezések esetén?
12. Milyen algoritmussal kereshetünk rendezett állományban?
13. Milyen hasításokat különböztetünk meg a kosarak száma szerint?
14. Mit nevezünk statikus hasításnak?
15. Mit nevezünk dinamikus hasításnak?
16. Milyen dinamikus hasításokat ismerünk?
17. Milyen szempontok alapján kell hasító függvényt választanunk (2 szempont)?
18. Mikor jó egy hasító függvény és ilyenkor milyen hosszúak a blokkláncok?
19. Ha túl nagynak választjuk K-t hasításkor, akkor ez milyen problémát okozhat?
20. Milyen keresésre nem jó a hasító indexelés?
21. Lineáris hasító index esetén mikor nyitunk meg új kosarat?
22. Mire használhatjuk a hasító táblákat az adatbázisban?
23. Mit nevezünk elsődleges indexnek?
24. Mit nevezünk másodlagos indexnek?
25. Mit nevezünk sűrű indexnek?
26. Mit nevezünk ritka indexnek?
27. Mit hívunk fedőértéknek?
28. Mit nevezünk többszintű indexnek?
29. Mit jelent a B+ fában a „B”?
30. Hány kulcsot és mutatót tartalmaz egy B+ fa csúcs?
31. Hogyan érdemes tárolni az ismétlődő értékeket a B+ fában?
32. Milyen költséget veszünk figyelembe a fizikai operátorok esetén?
33. Milyen relációs algebrai műveleteket vezethetünk vissza rendezésre?
34. A rendezés milyen két fajtáját különböztetjük meg?
35. Mennyi a belső rendezés IO költsége?
36. Külső összefésülő rendezésnél mit csinál a rendezési (sort) lépés?
37. Külső összefésülő rendezésnél mit csinál az összevonási (merge) lépés?
38. Külső összefésülő rendezéskor mikor kell több menetben végezni az összevonási lépést?

39. A vetítés milyen három lépés megvalósításából áll?
40. Az összekapcsolásoknak milyen megvalósításait használjuk (5 algoritmus)?
41. Az összekapcsolásoknál melyik táblát választjuk külső táblának általában?
42. Milyen esetekben érdemes a sort-merge összekapcsolást használni?
43. Milyen típusú összekapcsolásokra nem használható a hash-join?
44. Mi a lekérdezések optimalizálásának a célja és miket használunk fel ehhez?
45. Mi az a relációs algebrai kifejezésfa?
46. Milyen költségmodellt használunk relációs algebrai optimalizálás esetében?
47. Milyen módszert használ a szabály alapú (relációs) optimalizáció?
48. Miért mondjuk, hogy az eljárás heurisztikus relációs algebrai optimalizálás esetén?
49. Miért nem egyértelmű az eredmény relációs algebrai optimalizálás esetén?
50. A relációs algebrai kifejezésfában melyek az unáris csúcsok?
51. A relációs algebrai kifejezésfában melyek a bináris csúcsok?
52. A relációs algebrai kifejezésfában mik a levélcsúcsok?
53. Mik azok a relációs algebrai ekvivalencia szabályok?
54. Miért érdemes hamarabb szelektálni relációs algebrai optimalizálás esetén?
55. Miért érdemes természetes összekapcsolásokat képezni szorzások helyett relációs algebrai optimalizálás esetén?
56. Miért érdemes az unáris műveletek összevonni relációs algebrai optimalizálás esetén?
57. Miért érdemes a közös részkifejezéseket megkeresni relációs algebrai optimalizálás esetén?
58. Hogyan néz ki a bal-mély összekapcsolási fa?
59. Hogyan néz ki a bozótszerű összekapcsolási fa?
60. Milyen típusú összekapcsolási fákat érdemes vizsgálni?
61. Mit nevezünk az adatbázisban tranzakciónak?
62. Mi az ACID betűszó egyes betűinek a jelentése?
63. Mit jelent a tranzakció ACID-ból az A-hoz tartozó tulajdonsága?
64. Mit jelent a tranzakció ACID-ból az C-hez tartozó tulajdonsága?
65. Mit jelent a tranzakció ACID-ból az I-hez tartozó tulajdonsága?
66. Mit jelent a tranzakció ACID-ból az D-hez tartozó tulajdonsága?
67. A tranzakciók melyik tulajdonságát biztosítja a naplózás?
68. A tranzakciók melyik tulajdonságát biztosítja a konkurenciakezelés?
69. Milyen típusú hibák fordulhatnak elő?
70. Mit tekinthetünk adatbáziselemnek?
71. A tranzakció és az adatbázis kölcsönhatásának milyen három fontos helyszíne van?
72. Mit jelent az INPUT(X) művelet?
73. Mit jelent az OUTPUT(X) művelet?
74. Mit jelent a READ(X,t) művelet?

75. Mit jelent a WRITE(X,t) művelet?
76. Miből áll a napló?
77. Milyen naplázási megközelítéseket ismerünk?
78. Mi az UNDO naplázás hátránya?
79. Mi a REDO naplázás hátránya?
80. Mit jelent a WAL elv?
81. Milyen naplázást használ az Oracle?
82. Mi a különbség az ARCHIVELOG és NOARCHIVELOG működés között?
83. Mik azok a rollback szegmensek az Oracle adatbázisban?
84. Mit jelent a naplázás naplázása?
85. Milyen két típusa van az adatbázis mentésének?
86. Mit hívunk ütemezésnek?
87. Mit hívunk soros ütemezésnek?
88. Mi a soros ütemezés hátránya?
89. Mikor konfliktus-ekvivalens két ütemezés?
90. Milyen két megközelítése van az ütemezések sorbarendezhetőségének az elérésére?
91. Az aktív módszer milyen eszközöket használhat a sorbarendezhetőség elérésére?
92. Mi az a holtpont?
93. Milyen két megközelítése van a holtpont kezelésének?
94. Mit jelent az éhezés a tranzakciók esetén?
95. Milyen típusú zárákat használunk többmódon zárolás esetén?
96. Milyen elemeit zárolhatjuk az adatbázisnak többféle szemcsézettségű zárral?
97. Mi az a nem ismételhető olvasás?
98. Mik azok a fantom sorok?
99. Milyen optimista módjai vannak a sorbarendezhetőség elérésének?

## Tételek

1. Fizikai tárolás (másodlagos tárolók, RAID, lift algoritmus, pufferkezelő, lapcsere algoritmusok).
2. Adattárolás (blokkok, fájlok, fájlszervezés, rekordok felépítése, rendszerkatalógusok, rekord azonosítók).
3. Kupac és rendezett fájlszervezés, hasítótáblák.
4. Indexek (sűrű, ritka, elsődleges, másodlagos, több szintű, bittérkép index).
5. B+ fák.
6. Műveletek költségei, méretbecslés (kiválasztás, vetítés, rendezés, halmazműveletek).
7. Összekapcsolások költségei, méretbecslés.
8. Relációs algebrai ekvivalencia szabályok.
9. Szabály alapú optimalizáció.
10. Többtáblás lekérdezések optimalizációja.
11. UNDO naplázás.
12. REDO naplázás.
13. UNDO/REDO naplázás.
14. Ütemezések (soros, sorbarendezhető, konfliktusok, megelőzési gráf stb.).
15. Zárak (jogszerűség, konzisztencia, 2PL, várakozási gráf, holtpont).
16. További zármódok (osztott, kizártlagos, növelési, módosítási, kompatibilitási mátrix, felminősítés).
17. Zárolási ütemező, hierarchikus zárak, időbélyegzés és érvényesítés (zártábla, faprotooll, szemcsézettség, figyelmeztető zárak).

## Tételek (minta kérdésekkel)

1. **Fizikai tárolás (másodlagos tárolók, RAID, lift algoritmus, pufferkezelő, lapcsere algoritmusok)**
  - a. Hogyan működik a RAID X. szint?
  - b. Hogyan működik a helyreállítás paritásblokk használata esetén?
  - c. Hogyan lehet optimalizálni a blokkolvasásokat lemez esetén? Lift algoritmus működése.
  - d. Milyen lapcsere algoritmusok vannak, hogyan működnek?
2. **Adattárolás (blokkok, fájlok, fájlszervezés, rekordok felépítése, rendszerkatalógusok, rekord azonosítók)**
  - a. Hasonlítsa össze a kupac és rendezett fájlszervezést (előnyök, hátrányok).
  - b. Milyen memóriahez férés igazítás (word-alignment) módszereket ismer? Hogyan működnek?

- c. Hogyan lehet tárolni a rögzített hosszúságú rekordokat a blokkokban?  
Hogyan lehet kezelni a törléseket?
- d. Hogyan tároljuk a változó hosszúságú mezőket a rekordokban, és a rekordokat a blokkokban?

### **3. Kupac és rendezett fájlszervezés, hasítótáblák.**

- a. Mennyi a keresési idő kupac és rendezett fájlszervezés esetén? Hogyan működik a keresés?
- b. Hogyan működik a beszúrás rendezett állományba? Milyen megközelítések vannak?
- c. A (láncolt) statikus hasítás működése (beszúrás, törlés), példával.
- d. Kiterjeszthető hasító tábla működése (beszúrás, törlés, előnyök, hátrányok stb.), példával.
- e. Lineáris hasító tábla működése (beszúrás, törlés, előnyök, hátrányok stb.), példával.

### **4. Indexek (sűrű, ritka, elsődleges, másodlagos, több szintű, bittérkép index)**

- a. Hogyan keresünk sűrű indexben (algoritmus, költség)?
- b. Hogyan keresünk ritka indexben (algoritmus, költség)?
- c. Hogyan működnek a többszintű indexek (szintek száma, keresés költsége)?
- d. Bittérkép indexek működése (elkészítés, használat, hasznosság).

### **5. B-fák**

- a. A B+ fa csúcsainak (gyökér, köztes, levél) jellemzői.
- b. Hogyan keresünk B+ fában (példával)?
- c. Beszúrás B+ fába, példával (levél és köztes csúcs osztás).
- d. Törlés működése a B+ fában, példával.

### **6. Műveletek költségei, méretbecslés (kiválasztás, vetítés, rendezés, halmazműveletek)**

- a. Add meg a konjukciós összetett kiválasztás kiszámítási módjait.
- b. Add meg a diszjunkciós összetett kiválasztás kiszámítási módjait.
- c. A kiválasztás művelet méretbecslése (egyenlőségi feltétel, intervallumra vonatkozó feltétel, összetett kiválasztás).
- d. Add meg a vetítés költségét és méretbecslését.
- e. Külső összefésülő rendezés algoritmusa.
- f. Külső összefésülő rendezés költsége.
- g. Add meg az unió/különbség/metszet költségét és kimenet méretét (sorok száma, blokkok száma).

### **7. Összekapcsolások költségei, méretbecslés.**

- a. Egyszerű nested loops algoritmusa és költsége.
- b. Block nested-loop algoritmusa és költsége.
- c. Index nested-loop join algoritmusa és költsége.
- d. Sort-merge join algoritmusa és költsége.

- e. Hash-join algoritmusa és költsége.
- f. Összekapcsolások méretbecslése (három eset).

**8. Relációs algebrai ekvivalencia szabályok.**

- a. Mikor mondjuk azt, hogy két relációs algebrai kifejezés ekvivalens?
- b. Adjuk meg a ... szabályt! (az összes szabály szerepelhet!)

**9. Szabály alapú optimalizáció.**

- a. Melyek a szabály alapú optimalizálás heurisztikus alapelvei?
- b. Add meg a szabály alapú optimalizációs algoritmus lépéseiit.
- c. Mutass részletes példát a szabály alapú optimalizáció lépéseire.

**10. Többtáblás lekérdezések optimalizációja.**

- a. Milyen feladatként értelmezhetjük az összekapcsolási sorrend kiválasztását? Mi a lényege?
- b. Mik azok az összekapcsolási fák? Milyen típusú összekapcsolási fákat rajzolhatunk?
- c. Hány féle összekapcsolási fa lehetséges N tábla esetén?
- d. Hogyan működik a mohó algoritmus az összekapcsolási sorrend kiválasztására?
- e. Mi az Selinger-féle optimalizálás?

**11. UNDO naplázás.**

- a. Add meg az UNDO naplázás U1 és U2 szabályát.
- b. Mi a lemezre írás sorrendje UNDO naplázás esetén?
- c. Add meg a helyreállítás algoritmusát UNDO naplázás esetén.
- d. Egyszerű ellenőrzőpont képzés lépései (ellenőrzőpont leállítással).
- e. Működés közbeni ellenőrzőpont képzés UNDO esetén, helyreállítás.

**12. REDO naplázás.**

- a. Add meg a REDO naplázás R1 szabályát.
- b. Mi a lemezre írás sorrendje REDO naplázás esetén?
- c. Add meg a helyreállítás algoritmusát REDO naplázás esetén.
- d. Mi a különbség a REDO és a módosított REDO napló között?
- e. Ellenőrzőpont képzés REDO esetén, helyreállítás.

**13. UNDO/REDO naplázás.**

- a. Add meg az UNDO/REDO naplázás UR1 szabályát.
- b. Add meg az UNDO/REDO naplázás UR2 szabályát.
- c. Add meg a helyreállítás algoritmusát UNOD/REDO naplázás esetén.
- d. Ellenőrzőpont képzés UNOD/REDO esetén, helyreállítás.

**14. Ütemezések (soros, sorbarendevezhető, konfliktusok, megelőzési gráf stb.).**

- a. Mit nevezünk konfliktusnak?
- b. Mit nevezünk sorbarendevezhető és konfliktus-sorbarendezhető ütemezésnek?
- c. Mi az a megelőzési gráf?
- d. Hogyan működik a megelőzési gráf, mire használjuk?

**15. Zárok (jogszerűség, konzisztencia, 2PL, várakozási gráf, holtpont).**

- a. Mit jelent a jogsszerűség, konzisztencia?
- b. Mit nevezünk kétfázisú zárolásnak?
- c. A kétfázisú zárolás tétele.
- d. Mi az a várakozási gráf?
- e. Hogyan segít a várakozási gráf a holtpont felismerésében?
- f. Mit nevezünk a csúcsok topologikus sorrendjének?

**16. További zármódok (osztott, kizárolagos, növelési, módosítási, kompatibilitási mátrix, felminősítés).**

- a. Hogyan működik a zárolás, ha két típusú zárunk van (osztott és kizárolagos)?
- b. Mi az a kompatibilitási mátrix, mire használhatjuk?
- c. Hogyan működik a zárok felminősítése?
- d. Mire jó a növelési művelet és a növelési zár?
- e. Mi az a módosító zár?

**17. Zárolási ütemező, hierarchikus zárok, időbélyegzés és érvényesítés (zártábla, faprotokoll, szemcsézettség, figyelmeztető zárok).**

- a. Hogyan működik a zárolási ütemező?
- b. Mi az a zártábla? Hogyan épül fel?
- c. Mit jelent a zárok szemcsézettsége?
- d. Hogyan működik a figyelmeztető protokoll?
- e. Mire használjuk a faprotokollt? Hogyan működik?
- f. Hogyan biztosítható a sorbarendezhetőség időbélyegzéssel?