

NYELVFÜGGETLEN PROGRAMOZÁS ÉS ADATBÁZISKEZELÉS

3. forduló



A kategória támogatója: SAP Hungary Kft.

RENDELKEZÉSRE ÁLLÓ IDŐ:

60:00

Ismertető a feladathoz

Fontos információk

Ezután a forduló után automatikusan jár a [kitartóknak szóló garantált ajándékunk](#), érdemes kitöltened a feladatlapot! :)

Ha kifutsz az adott feladatlap kitöltésére rendelkezésre álló időből, a felület **automatikusan megpróbálja beküldeni** az addig megadott válaszokat.

A kérdésekre **mindig van helyes válasz**, olyan kérdés viszont nincs, amelyre az összes válasz helyes!

Egyéb információkat a [versenyszabályzatban](#) találsz!

Felhasznált idő: 01:46/60:00

Elért pontszám: 0/26

1. feladat 0/6 pont

Melyik állítás helytelen az adatbázistábla elsődleges kulcsával kapcsolatban?

Válasz

- ☐ Elképzeltető, hogy az elsődleges kulcs csak egy mezőből áll.
- ☐ Előfordul, hogy egy másik adatbázistábla idegen kulcsa az adott adatbázistábla elsődleges kulcsára utal
- ☐ Az elsődleges kulcs egyedileg azonosítja az adatbázistábla minden rekordját
- ☒ Az elsődleges kulcs mezőjének (mezőinek) numerikus típusúnak kell lennie
- ☐ Bizonyos esetekben a Globally Unique Identifiert (GUID) elsődleges kulcsként alkalmazzák
- ☐ Lehetséges, hogy az elsődleges kulcs egynél több mezőt tartalmaz

Magyarázat

Az elsődleges kulcs mezőjének (mezőinek) numerikus típusúnak kell lennie

2. feladat 0/10 pont

Cache kihagyás

Három blokk hosszúságú gyorsítótárad van, ami az elején üres. A gyorsítótár-stratégia Least Recently Used (LRU). Hány cache kihagyást (cache miss) okoz a következő program?

```
FOR i:=1 TO 5
  READ_BLOCK_N( (i*2) MOD 5 )
  READ_BLOCK_N( i MOD 5 )
ENDFOR
```

Válaszok

- ☐ 3
- ☐ 4
- ☒ 5
- ☒ 6
- ☐ 7
- ☐ egyik sem a fentiek közül

Magyarázat

A Least Recently Used (LRU, legrégebben használt) gyorsítótár-stratégia alkalmazása esetén, ha a cache-ben már nincs hely az újonnan tárolandó elemnek, a legrégebben használt elem kerül eltávolításra.

A program lefutásának szemléltetése:

```
I=1
-> Cache tartalom: -, -, -
READ_BLOCK_N(2): Cache MISS
-> Cache tartalom: 2, -, -
READ_BLOCK_N(1): Cache MISS
-> Cache tartalom: 1, 2, -

I=2
READ_BLOCK_N(4): Cache MISS
-> Cache tartalom: 4, 1, 2
READ_BLOCK_N(2): Cache HIT
-> Cache tartalom: 2, 4, 1

I=3
READ_BLOCK_N(1): Cache HIT
-> Cache tartalom: 1, 2, 4
READ_BLOCK_N(3): Cache MISS
-> Cache tartalom: 3, 1, 2

I=4
READ_BLOCK_N(3): Cache HIT
-> Cache tartalom: 3, 1, 2
READ_BLOCK_N(4): Cache MISS
-> Cache tartalom: 4, 3, 1

I=5
READ_BLOCK_N(0): Cache MISS
-> Cache tartalom: 0, 4, 3
READ_BLOCK_N(0): Cache HIT
-> Cache tartalom: 0, 4, 3
```

Összesítve:

Cache HIT (találat): 4

Cache MISS (kihagyás): 6

A pszeudokód nem magyarázza el, hogy a "FOR i:=1 TO 5" sorban az 5 az exkluzív vagy inkluzív. Ennek megfelelően az 5 megoldást is elfogadtuk.

3. feladat 0/10 pont

Piramis

A pozitív egész számokat egy piramisba rendezzük a végtelenségig.

```
1
2 3
4 5 6
7 8 9 10
```

Az első 25 kettő hatványra (2 a nulladikon, 2 az elsőn, 2 a másodikon stb.) adjuk össze az ő sor- és indexszámaikat (az indexelést és a hatványozást is nulláról kezdjük). **Mennyit kapunk?**

Válaszok

A helyes válasz:

28556

28581

40842

162439

4628700086782023

33574155

Sorszámok összege: 19774 Indexek összege: 33554406

Magyarázat

A feladatot többféleképp lehetett értelmezni, ezért többféle megoldás elfogadott.

[Legfontosabb tudnivalók](#)

[Kapcsolat](#)

[Versenyszabályzat](#)

[Adatvédelem](#)

© 2022 Human Priority Kft.

KÉSZÍTETTE

Megjelenés

 Világos 