

WUOLAH



BinTrack2

www.wuolah.com/student/BinTrack2



1850

Recopilacion 2º parcial VyF.pdf

Preguntas de VyF



2º Fundamentos de Bases de Datos



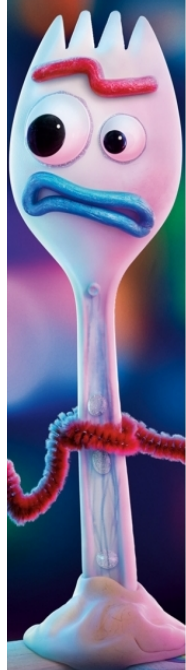
Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
UGR - Universidad de Granada

Recopilación preguntas 2º parcial FBD Preguntas de Verdadero y Falso.

1. Conociendo el RID de un registro no hace falta más que un acceso a disco para recuperarlo. (V)
2. Conviene que estén relacionados el tamaño de los bloques físicos y el de las paginas para mejorar el rendimiento del sistema de almacenamiento (V)
3. Cuando la clase de un índice es compuesta (C1, C2) resulta eficiente el uso del índice para buscar por C1 o por C2.
4. El acceso directo a bloques o cubos produce menos lecturas en disco que el acceso directo a registros. (V)
5. El acceso directo a registros garantiza siempre que encuentre una tupla con una sola lectura de bloque (F)
6. El acceso directo a registros no permite realizar la lectura secuencial de datos en un rango (V)
7. El administrador de la base de datos puede decidir la forma de agrupamiento en páginas de los archivos que corresponden a las tablas de una base de datos (V)
8. El agrupamiento por defecto en el nivel interno es intra-archivo (V)
9. El gestor de disco forma parte del SGBD (F)
10. El hashing dinámico es el método de acceso que mejor distribuye los datos en el disco y, por tanto, el que menos desperdicio ocasiona (V)
11. El hashing dinámico es muy eficaz porque la tabla hash no está en la memoria principal (F)
12. El hashing dinámico es muy eficaz porque la tabla hash va en memoria principal (F)
13. El hashing dinámico hace falta una estimación del número de datos a insertar para cada tabla hash (V)
14. El índice denso es adecuado para consultas por rango de valores del campo clave (V)
15. El índice denso no es rentable cuando se actualiza o se inserta con mucha frecuencia (V)
16. El índice denso ocupa el mismo tamaño que el propio fichero que indexa (F)
17. El índice no denso es el único mecanismo de indexación posible cuando los datos están ordenados físicamente (F)
18. El índice no denso es mucho menor que el denso cuando caben varios registros en un bloque (V)
19. El índice no denso mejora el barrido ordenado completo del fichero por la clave primaria (F)
20. El índice no denso mejora el barrido ordenado completo del fichero por la clave física. (F)
21. El índice no denso permite realizar preguntas de tipo existencial sin acceder al fichero de datos (F)
22. El mantenimiento de un índice no denso es menos costoso que el de un índice denso (V)
23. El nivel interno de una base de datos está enteramente gestionado por el S.O del ordenador(F)
24. El número de accesos a disco que hacen falta para obtener una página depende del tamaño de la página y del tamaño del bloque físico.
25. El orden de un árbol B fija el número de punteros que salen de un nodo. (V)
26. El orden de un árbol está determinado por el tamaño de la página que se asigna a los nodos del árbol (V)
27. El orden de un árbol influye directamente en el número de niveles (V)
28. El rendimiento de un índice no denso desciende considerablemente cuando se realizan inserciones o borrados (F)
29. En acceso directo a registros, si se produce una colisión, habrá un hueco en el fichero maestro que nunca se va a aprovechar. (V)



30. En el agrupamiento inter-archivo se ubican en una página registros de distinto tipo. (V)
31. En el hashing extendido lo mejor es que la pseudollave se ajuste el tamaño del índice que se guarda en memoria (V)
32. En el hashing extendido lo mejor es que la pseudollave tenga muchos dígitos (F)
33. En el hashing extendido no se producen desbordamientos (F)
34. En el hashing extendido una mala elección en el tamaño de las páginas puede obligar a reorganizar completamente la estructura. (F)
35. En el nivel interno de una BD hay que tener en cuenta también el nivel físico que gestiona el acceso a fisco (V)
36. En ficheros no ordenados físicamente, no se pueden montar índices no densos (V)
37. En hashing dinámico hace falta una estimación del número de datos a insertar para dimensionar la tabla hash (V)
38. En un archivo almacenado puede haber más de un índice primario (F)
39. En un índice denso el número de elementos es el mismo que el del archivo principal.
40. En un índice multinivel el índice de primer nivel (nodo hoja) puede ser denso o no denso (V)
41. En una base de datos puede haber más de un índice primario. (F)
42. En una estructura hash dinámica, al insertar un elemento que supere el tamaño de un cubo, siempre se producirá desbordamiento tanto en ese cubo como en el directorio (F)
43. En una organización secuencial no es necesario que los registros mantengan ningún orden en peculiar (F)
44. La actualización de los archivos puede no influir en la actualización de los índices no densos (V)
45. La clave de una tabla organizada por índice puede estar definida sobre cualesquiera de sus campos (F)
46. La organización multilistas puede servir para conectar fichero y es la base de datos basados en grados. (V)
47. Las actualizaciones en los archivos almacenados no tienen por qué determinar la actualización de los índices no densos. (V)
48. Las consultas basadas en OR sobre campos indizados mediante índices bitmaps, obtienen las tuplas que satisfacen la condición directamente de los índices (V)
49. Las páginas que componen un archivo almacenado no tienen por qué estar consecutivas en disco (V)
50. Las sentencias CREATE TABLE y CREATE INDEX de SQL generan nuevos conjuntos de páginas (archivos almacenados) en el nivel interno (V)
51. Lo normal es que cada archivo almacenado del nivel interno se almacene en un fichero físico separado (F)
52. Los árboles B se montan en memoria para no tener que acceder a disco más que una vez para llegar a un registro (F)
53. Los bloques usados para almacenar los datos de la BD pueden ser de distinto tamaño dependiendo del tamaño de los registros que se almacenen en ellos (F)
54. No se pueden resolver consultas basadas en AND sobre dos campos indizados mediante índices bitmaps, usando estos índices (F)
55. Para búsquedas basadas en el valor de la clave, los árboles B garantizan el acceso a un número fijo de páginas de base de datos.
56. Para montar un índice denso, los registros tienen que estar ordenados físicamente por algún campo. (F)
57. Puesto que es una variante de índice no denso, solo se puede montar un árbol B (sobre la clave física) de un archivo (F)
58. Se puede combinar una organización multilistas con un árbol B para gestionar los accesos en una estructura de datos jerárquica. (F)
59. Se puede montar tantos índices densos como se necesiten (V)
60. Se puede montar un árbol B sobre cualquier campo clave utilizando un índice denso como conjunto secuencia. (V)
61. Se pueden montar tantos índices densos como se necesiten (V)

- 62. Se pueden montar tantos índices no densos como sea necesario **(F)**
- 63. Si se cuántos registros va a tener el archivo almacenado y cuantos valores distintos de la clave puedo dimensionar adecuadamente el acceso directo a cubos **(F)**
- 64. Todas las páginas de una base de datos tienen la misma estructura. **(F)**
- 65. Un factor de bloqueo mayor a 1 implica tener más de un registro por página **(V)**