BSCCS2001: Mock Quiz 3 Mock with Solutions
Weeks 7 to 9

1. Match the following services as provided by the (A) Web Browsers, and the (B) Web Servers.
   The services are:

   1. It requests for a specific resource using HTTP or its secure variant HTTPS.
   2. It accepts requests via HTTP or its secure variant HTTPS and responds with the content of that resource or an error message.
   3. It has a rendering engine that displays the retrieved webpage on the user's device.
   4. When it receives a request which is for an executable program, it executes the program, and sends back the HTML document that is generated.
   5. It can accept and store resources sent from the user agent

   [ARUP: MCQ: 3 points]

   √ (A)-1, 3, (B)-2, 4, 5

   ◯ (A)-1, 2, 3, (B)-4, 5

   ◯ (A)-2, 3, (B)-1, 4, 5

   ◯ (A)-2, 4, 5 (B)-1, 3

Solution: A Web Browser provide the following services:

- It requests for a specific resource using HTTP or its secure variant HTTPS.
- It has a rendering engine that displays the retrieved webpage on the user's device.

A Web Servers provide the following services:

- It accepts requests via HTTP or its secure variant HTTPS and responds with the content of that resource or an error message.
- When it receives a request which is for an executable program, it executes the program, and sends back the HTML document that is generated.
- It can accept and store resources sent from the user agent

2. Consider a bitmap index, on the attribute *designation* of a table **Faculties**, with the following properties:

   - The attribute *designation* allows only the following values as entries: 'Assistant Professor', 'Associate Professor', and 'Professor'.
   - The number of records is 24,000.

   What is the size of the bitmap index? [ARUP: MCQ: 2 points]

   ◯ 24,000 bytes

   ◯ 3 bytes

   √ 9,000 bytes

   ◯ 3,000 bytes

Solution: A bitmap index on an attribute, having k distinct values and n records, has size $s = n \times k$ bits.
Thus, the size of bitmap index is $24,000 \times 3 = 72,000$ bits or $9,000$ bytes.

Consider the given instance of a B^+-tree (in Figure 1) with order 4 and answer the questions 3.

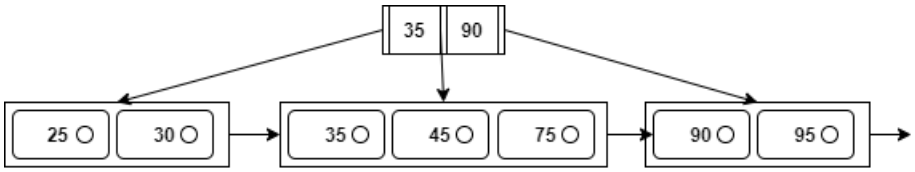


Figure 1: Table **Students**

3. Perform the following operations (in the given order) in the instance given in Figure 1.

   - Insert key 10
   - Insert key 60
   - Insert key 20

   Identify the correct set of keys in the root node of the resultant B^+-tree. [ARUP: MSQ: 4 points]

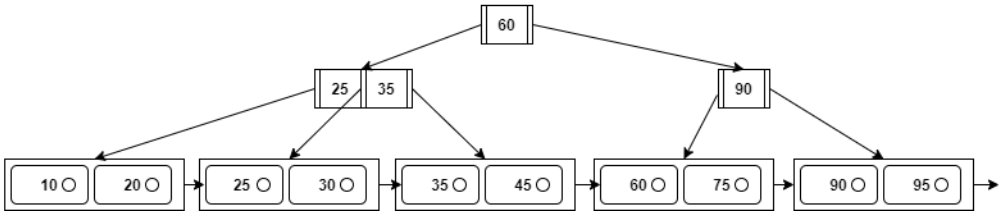
   ◯ {35, 60, 90}

   ◯ {25, 35, 90}

   √ {60}

   ◯ {35, 60}

Solution: The resultant B^+-tree in Figure 3.



Thus, the root is 60.

4. Consider the table **Students** in Figure 2, and answer the following question.

Name	Age	Country	Score
Tom	13	Australia	70
Lucy	15	Scotland	95
Frank	16	Germany	76
Jane	13	Australia	49
Robert	16	Germany	93
Ryan	18	Ireland	56
Mike	13	Germany	84

Figure 2: Table **Students**

Fill in the blanks in the Python program below (as per the instruction given as comments) such that it alters the *score* of **Jane** to 60. [ARUP: Programming: 4 points]

```
import psycopg2
def updateRecord(name, score):
    conn = None
    try:
         conn = psycopg2.connect(database = "mydb", user = "myuser", \
                        password = "mypass", host = "127.0.0.1", port = "5432")
        cur = _____________     # LINE-1: create a new cursor             (i)
        # LINE-2: create SQL statement with passing
        # the parameters with %s - name, score
        cur.execute(_____________________________________, ___________) (ii a) (ii b)
        _____________           # LINE-3: commit the changes to the database (iii)
        # LINE-4: show the number of rows updated
        # (applicable for any instance of the database)
        print ("Total number of rows updated :", _____________)       (iv)
        cur.close()
    except (Exception, psycopg2.DatabaseError) as error:
        print(error)
    finally:
        if conn != None:
            conn.close()
updateRecord('Jane', 60)
```

Note that:

- This is not an auto-graded question.
- (ii a) corresponds to the blank for the SQL statement
- (ii b) corresponds to the blank for the parameter to be passed to the SQL statement

- The answer has to be entered in the format given below:
  (i)
  (ii a)
  (ii b)
  (iii)
  (iv)

Solution:

```
(i): conn.cursor()
(ii): "update Students set score = %s where name = %s", (score, name)
(iii): conn.commit()
(iv): cur.rowcount
```

Suppose a data file has 1,00,000 **EMPLOYEE** records. Each record has the following fields: EID (16 bytes), NAME (24 bytes), ADDRESS (64 bytes) and PHONE (10 bytes). We also consider extra 14 bytes per record to keep additional information such as deletion marker. Consider that the records are unspanned and of fixed length. Block size is 1 KB. We consider a secodary index constructed on the attribute NAME (which is a non-ordering candidate key field) of the data file. The size of each block pointer is 8 bytes. Based on this information, answer the questions from 5 to 8.

5. How many blocks are required to store the data file? [Arup: NAT: 2 points]
Answer: 12500

> **Solution:** Each record size, $R = 16 + 24 + 64 + 10 + 14 = 128$ bytes.
> Number of records per block is $= \lfloor \frac{1 \times 2^{10}}{128} \rfloor = 8$.
> Thus, total number of blocks are required for the data file, $b = \lceil \frac{1,00,000}{8} \rceil = 12,500$.

6. How many blocks are required to store the index? [Arup: NAT: 2 points]
Answer: 3125

> **Solution:** Size of each entry in index is = [size of key field] + [size of block pointer] $= 24 + 8 = 32$ bytes (since size of NAME is 24 bytes).
> Thus, number of entries per block is $= \lfloor \frac{a \times 2^{10}}{32} \rfloor = 32$.
> Since we use secondary index on a non-ordering candidate key NAME, it must be a dense index with an entry for each record stored in the data file. The total number of blocks are required to store the index is $= \lceil \frac{1,00,000}{32} \rceil = 3125$.

7. How many blocks need to be accessed, on an average, for searching a key on the data file? [Arup: NAT: 2 points]
Answer: 6,250

> **Solution:** From the solution of 5, the data file is accommodated in $b = 12,500$ blocks. Thus, on an average, the number of blocks to be accessed to perform a linear search is $\frac{b}{2} = \frac{12,500}{2} = 6,250$ blocks.

8. How many blocks need to be accessed, on an average, using a binary search on the index file to search and access a record? [Arup: NAT: 2 points]
Answer: 13

Solution: From the solution of 6, the index is stored in 3125 blocks. Thus, on an average, the number of blocks to be accessed to search the key is $\lceil log_2(3125) \rceil = 12$. To access the record, we need to access the block containing the record from the data file. Thus, number of block access required to access the record is $= 12 + 1 = 13$.

9. *It is a cross platform interface which represents the structure of an HTML/XML document in the form of a logical tree.*

   Which one of the following is the correct choice for the definition given above?

   [Bhaskar: MCQ: 2 points]

   - √ Document Object Model
   - ◯ Hypertext Tree Model
   - ◯ Document Tree Model
   - ◯ None of the above

10. Consider a system with block size of 16 KB. A B^+ tree is implemented for indexing a database on this system. The search key field is 8 Bytes and the size of child pointer is 4 Bytes also the size of a record pointer is 16 Bytes.

    Consider there are 5 million $(5 * 10^6)$ records in a table to be indexed. If unspanned organization is used, what should be the minimum number of disk accesses required to fetch any record from the secondary memory assuming the tree is mostly full?

    [Bhaskar: NAT : 4 points]

    Ans: 4

    Solution:
    Let us assume that the order of the B^+ tree is n.
    There will be n child pointers and n-1 key values inside one internal node of the tree.
    Also a single node must fit inside one block, therefore

    $$(n-1) * 8 + n * 4 \leq 16 * 2^{10}$$

    Or $n = 1366$
    Number of disk accesses required to search the record which matches with the given key $= \lceil \log_{1366} 5000000 \rceil = 3$
    To fetch the particular record from secondary memory one more disk access is required.
    Therefore the naswer is **4** disk accesses.

11. Consider a file of 200 MB stored on a hard disk of 500 GB. Seek time of the hard disk's read head is 5 ms, rotational speed is 10000 RPM. The disk has 100 sectors/track and sector size is 256 Bytes.
What is the transfer rate (in Bytes/ms)?

[Bhaskar: MCQ: 4 points]

- ◯ 8533.34 Bytes/ms
- ◯ 2560 Bytes/ms
- ◯ 3656.67 Bytes/ms
- √ None of the above

Solution:
Transfer rate is the rate at which data is read from the disk. It can be calculated as follows.
The given rotational speed of the disk = 10,000 RPM.
i.e. disk rotates 10,000 times in 60 sec.
So, time required for making 1 rotation = $60/10,000 = 6 * 10^{-3} sec = 6$ ms

*Total number of bytes present on one track= number of sectors/track * sector size*
Total number of bytes present on one track= $100 * 256 = 25600$ bytes

Transfer Rate = Bytes on one track / time for making one rotation
Transfer Rate = $25600/6 = 4266.67$ Bytes/ms

Therefore option D is the correct choice.

12. What is the time complexity of the following code snippet?

[Bhaskar: MCQ: 3 points]

```
def fun1( x, y ):
    if x % y == 0:
        return y
    else:
        return fun1( y, x % y )
```

◯ O(x)

◯ O(y)

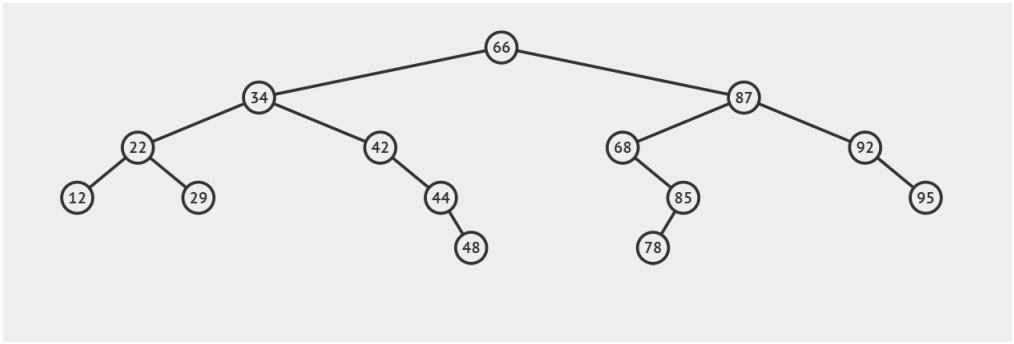
√ O($\log x$)

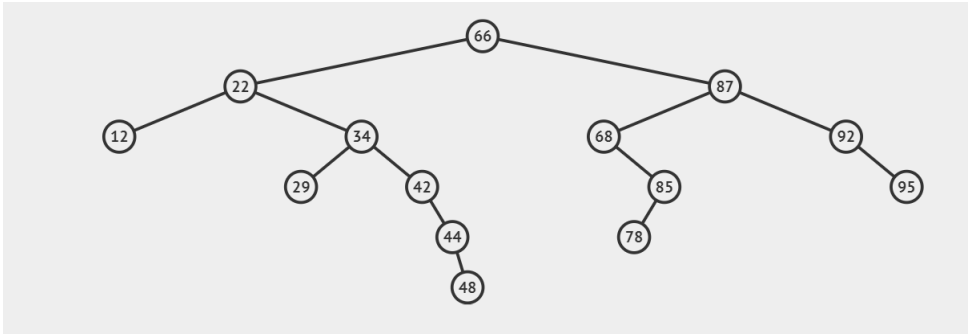
◯ O($y \log x$)

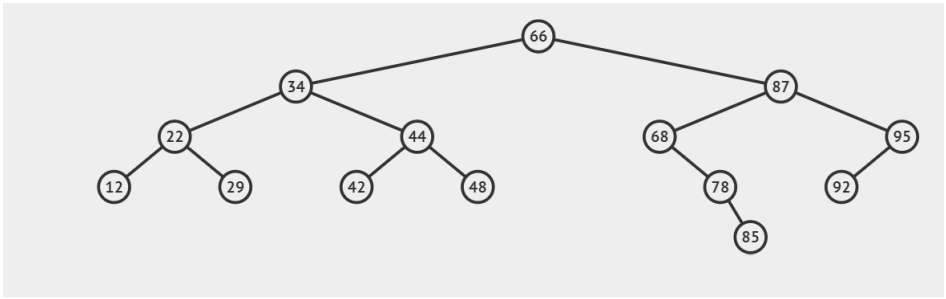
Solution:
Axiom 1: $x \bmod y < x/2$
This is a valid statement because $x \bmod y < x/2$ when $y > x/2$ and also when $y \leq x/2$.
Assume that the initial values are x_0 and y_0, then in the next iteration of the function the parameters will be x_1, y_1 such that:
$x_1 = y_0$ and $y_1 = x_0 \bmod y_0$
Likewise for subsequent iterations,
$x_2 = y_1 = x_0 \bmod y_0$ and $y_2 = x_1 \bmod y_1$
Generalising :
$y_{i+2} = x_{i+1} \bmod y_{i+1}$
Applying the axiom 1 we get:
$y_{i+2} = x_{i+1} \bmod y_{i+1} < x_{i+1}/2$
Therefore it is proven that the recursive calls would reduce the size of the larger number at least by half for every alternate iteration. Therefore the time complexity of the above function would be O(log x).

Will be discussed in the revision session.

13. Choose the correct Binary Search Tree for the following sequence:
66,87,34,95,22,68,44,78,12,92,29,85,48,42 [Piyush: MCQ: 2 points]

○ 

○ 

✓ 

○ None of the above

Solution: Option 1: It represents the BST for sequence
66,22,34,42,87,68,44,85,12,92,78,29,48,95
Option 2: It represents the BST for sequence
66,87,34,42,22,68,44,85,12,92,29,78,48,95
Option 3 : It represents the BST for sequence
66,87,34,95,22,68,44,78,12,92,29,85,48,42
Thus, option 3 is correct.

Consider the table **T20WC** as shown in Figure 3

Team	Matches	Won	Lost	Points
England	5	4	1	8
Australia	5	4	1	8
South Africa	5	4	1	8
Sri Lanka	5	2	3	4
West Indies	5	1	4	2
Bangladesh	5	0	5	0

Figure 3: Table **Students**

14. Assuming that we have constructed a bitmap index for all the columns. Which of the following statements is/are true? [Piyush: MSQ: 3 points]

◯ Bitmap index for **Team** and **Matches** will look same.

√ Bitmap index for **Won**, **Lost** and **Points** will have the same size but values are different.

√ Bitmap index for **Matches** will have 1 row.

◯ Bitmap index for **Team** will have 1 row.

Solution:

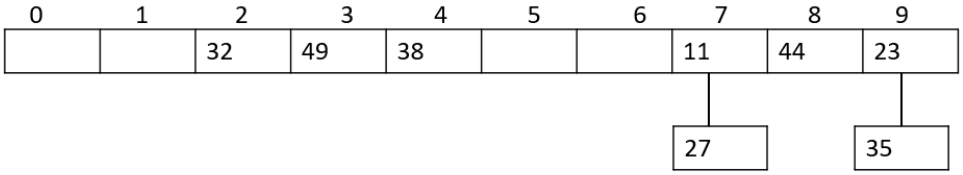
- Bitmap Index for **Team** will have 6 rows.
- Bitmap Index for **Matches** will have 1 row.
- Bitmap index for **Won**, **Lost** and **Points** will have the same size but values are different.

15. Consider a disk having 128 tracks per surface, 512 sectors per track and 256 bytes/sector. If the minimum number of bits required to access a sector is 21, then find out the number of platters required? [Piyush: NAT: 3 points]
Ans: 16

Solution: The total number of sectors = 2^n, where n is a number of bits required to access a sector.
Total no. of sectors = 2^{21} = 2097152.
Now, Total no. of sectors = no. of recording surfaces * 128 * 512 = 2097152
Thus, no. of recording surfaces = 32.
No of platter=16

16. Consider a hash table of size 10, with starting index 0 to 9, and a hash function is given by $h(n) = n^2 + 2n + 4(mod 10)$. Assuming the hash table is initially empty, when the sequence 11, 32, 23, 44, 27, 49, 35, 38 is inserted into the table using closed hashing, then how many empty locations are there in the hash table? [Piyush: NAT: 3 points]
Ans: 4

Solution: Hash function is given by $h(n) = n^2 + 2n + 4(mod 10)$
h(11) = 147 (mod 10) = 7,
h(32) = 1092 (mod 10) = 2,
h(23) = 579 (mod 10) = 9,
h(44) = 2028 (mod 10) = 8,
h(27) = 787 (mod 10) = 7,
h(49) = 2503 (mod 10) = 3,
h(35) = 1299 (mod 10) = 9,
h(38) = 1524(mod 10) = 4



So, there are 4 empty locations in the hash table.

17. A B^+ tree of order 3 is initially empty. then the following key values are inserted in the given order

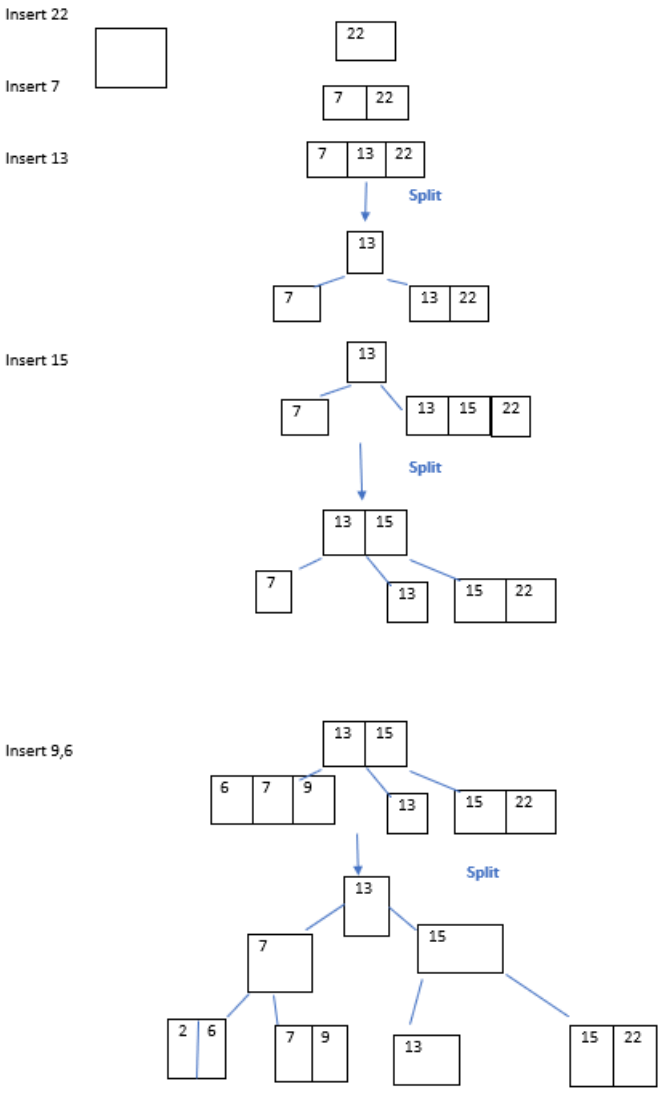
$$22, 7, 13, 15, 9, 6, 2$$

The number of times leaf nodes would get split up as a result of these insertions is

[Subendu: NAT: 3 points]

Answer: 3

Solution: The leaf node will split insertion of 13,15,6



18. In an MVC architecture, what is the function of the controller component? [Anjana: MCQ : 2 points]

- ◯ It is used for the UI logic of an application.
- ◯ It is responsible for all the data related logic that the user has to deal with.
- √ It acts as a request handler that handles the user requests.
- ◯ All of the above.

Solution: An MVC architecture separates an application into three components- Model, View and Controller.

1. The **Model** component is responsible for manipulating and interacting with the data of the database, basically all the data related logic that the user has to work with.
2. The **View** component is used for the UI logic of an application.
3. The **Controller** component acts as a request handler that handles the user requests. It uses the Model component to process all the business logic and incoming requests and manipulate the data, and then it interacts with the View component to give the final output.