COMP-3150 FINAL REVIEW CLASS: LECTURE NOTES (LAST CLASS) Covered on Dec. 3, 2020 or Apr. 6, 2021

COMP 3150 WINTER 2021 FINAL REVIEW CLASS:

Links to any posted recorded in-class lectures for Review class of Dec. 3, 2020 (similar to Apr. 6, 2021) will be found below:

1. Links on one drive to recorded lecture of Thursday, Dec. 3, 2020 (similar to Apr. 6, 2021) (saved in one recorded session) is provided below in sequences 1:

   Frist Recording 1 of 1:

   https://uwin365-my.sharepoint.com/:v:/g/personal/cezeife_uwindsor_ca/EXUgt97JipNNh32wXvASpRcBPSsgtxzzc8Lg4M2NM3fazg

***

This class has two parts:

1. Final Examination Reviews: A review and summary of all materials covered in this course in Winter 2021 term. The schedule of final examination, final exam viewing and final exam make-up are also provided.
2. Final Exam Practice Discussions: Some solutions to the final examination practice given to class earlier in April 6,

2021. This is in honor of those students who have attempted to solve all or some of these questions in preparation for the final exams.

Good Luck in you Finals !!

Part I: The Final Exam Reviews

Comp-3150-01 (60-315-01) Winter 2021 Final Exam Preparation

Upcoming Class Events

Final Exam Date (covers materials from chs. 1, 2, 3, 5, 14, 6, 7, 8, 17 of book), lecture slides and written lecture notes and examples in class.	**Wed. April 21, 2021 (08:30am): Time: 08:30am – 12:30pm. Where: (Venue:online through blackboard)**
Tentative Exam Viewing Date	Sat. Apr. 24, 2021 (10:00am to 4:00pm) online meeting with supervising GA/TA through blackboard collaborate or email.

***** GOOD LUCK ******

NOTE: In the event you cannot write the exam due to illness or some other emergency (approved CS medical certificate required for proof), call/send email to the CS office (csinfo@uwindsor.ca) or send email to me within one day or as soon as possible to inform. There is only one make-up exam for all, who could not write final and are pre-approved by me to write the make-up exam; that is on Friday, May. 14, 2021 (1:00 to 5:00 pm) ONLINE.

We hope to mount exam viewing sessions on Sat., Apr. 24, 2021 (10:00am to 4:00pm) online meeting with supervising GA/TA through blackboard collaborate or email, subject to we, completing the marking. Viewing is as scheduled unless otherwise announced.

Student Name and id:_____

These questions in the final exam practice are to help with preparing for final exams. To prepare well, read through all of the lecture slides and the notes you have been writing to ensure you feel comfortable with the materials taught in the course. Refer to the text book when you want more information about some concepts in the lecture slides.

Format of the final Exam: ~~2 hours 30 minutes~~ (this time 4 hours) with two Sections (A and B).
- Section A has 15 multiple choice questions for a total of 30 marks while Section B has 4 full questions worth a total of 70 marks.
- Check the final exam practice questions for help with preparation. Note that no solutions are provided online for the final exam practice questions. The only review to these questions are in class with students.

We have covered materials from Chs. 1, 2, 3, 5, 14, 6, 7, 8, and 17 of the course book and should know: Chs 1 – 3: DBMS concepts and architecture, data models based on entities and relationships, data base design using the Entitiy-Relationship model (ER model), page 83 of book and slide 19 of Chapter 3 notes provide a summary of notations used for representing a database in ER diagrams. We should be able to design a database using ER model, convert between ER models and relational model and vice versa.
Ch. 5: Relational data model concepts, relational database design and constraints. Some concepts are: Database schemas, database states, primary key, candidate key, superkey, foreign key, domains, attributes, relations, degree of a relation, cardinality of a relation, system catalog, Null values. Some constraints are domain, key, foreign key, referential integrity, and entity integrity. Relational model operations are those for creating and updating the schema and those for creating and

updating data and retrieving data. Those operations for data manipulation that may violate database constraints are Insert operation, delete operation, and update operations.

Ch. 14: Database design guidelines and theories of normal forms based on functional dependencies. For good database design, attributes from multiple entity types (e.g. student, course) should not be combined in one entity. Such design would cause base relations to not be in at least 3NF and suffer from anomalies due to redundant storage of data. Design should minimize the need for attributes that have a lot of NULL values. To minimize generation of spurious tuples (not in the original relations) from natural joins of relations, design relations that can be joined on primary key, foreign key attribute pairs. Functional dependencies (FDs) are constraints that are derived from the meaning and interrelationships of the data attributes. The FD: X → Y holds in a relation if whenever any two tuples have the same value for X, they must have the same value for Y. Normalization is the process of decomposing badly designed relations into smaller relations. Normal form uses keys and FDs to certify whether a schema is in a particular normal form such as 1NF, 2NF, 3NF and BCNF. Relations that are in 3NF and higher normal forms are considered to be normalized. 1NF allows relations to store only single values for each attribute of a tuple. A relation R is in 2NF if every non-key attribute is fully functionally dependent on the key (that is, the only determinants (X) of non-key attributes (Y) in the relation R, have no subsets of them (X) also determining any non-key attributes (Y)). A relation is in 3NF if it is in 2NF and no non-key attribute of R is transitively dependent on the primary key (that is, the only determinants (X) of non-key attributes (Y) in the relation R, have no other non-key attribute (Z) also determining any non-key attributes (Y)). A relation is in BCNF if whenever an FD X → Y holds in R, then X is a superkey of R (that is, the superkey (X) is the only determinant and no non-key attribute should determine any part of the superkey). While 3NF allows R that are in either the FD condition (a): X →A where X is a superkey or (b): A is a key attribute of R (that is, a non-key attribute X can determine part of a key); the BCNF does not allow condition (b). Fig. 14.12 on page 485 of book steps you through the LOTS relation in 1NF, 2NF, 3NF, while Fig 14.13 shows you its decomposition into BCNF.

Ch. 6: Basic SQL: for data definition language (DDL) and data manipulation language (DML). SQL DDL are for creating database schemas, database tables, modifying table schemas, deleting table schemas with constraints. SQL DML are creating, deleting and updating data and for data retrieval. The SQL DDL commands include: CREATE TABLE, DROP TABLE, ALTER TABLE commands. The SQL DML for data retrieval include the SELECT-FROM-WHERE command in spj form, nested form, with aggregate functions and GROUP BY, HAVING and ORDER BY clauses. The SQL DML for data modification include INSERT INTO, DELETE-FROM-WHERE, UPDATE-SET-WHERE statements.

Ch. 7: More details of SQL DML for more complex data retrievals (not just the select-project-join queries) including nested queries, aggregate functions (SUM, COUNT, AVG, MIN, MAX) with group by, having and order by clauses are discussed. SQL for creating views and more discussions of schema modifications using DROP TABLE, ALTER TABLE are discussed. Slides 36 and 37 of Ch. 7 notes has a summary of SQL commands and syntax.

Ch. 8: Relational algebra unary operations of (SELECT (σ), PROJECT (π) and RENAME(ρ)) for querying databases are discussed. The relational algebra binary operations of (JOIN ($\bowtie$) and DIVISION (-:-)) are discussed. The set operations of UNION (U), INTERSECTION ($\cap$), DIFFERENCE (MINUS) (-), CARTESIAN PRODUCT (CROSS PRODUCT)(x)) are discussed. Use of aggregate functions in relational algebra querying is discussed. The tuple relational calculus (TRC) for expressing retrieval queries in the form of {t | COND(t)} was also covered. Cond(t) can be a formula consisting of an atom such as Employee(t), t.salary > 30000; or an atom connected with logical operators (such as AND, OR, NOT); and also can include existential (e.g. ($\exists$ **d))** and universal quantifiers (e.g., ($\forall$ t)).

Ch. 17: Use of Index files for speeding up database records from secondary disk storage is discussed. General description of index file structures as alternative access paths to primary data file on disk is given. Single level ordered indexes store the secondary index file on disk as a file ordered on the index field where each index key also contains a second field of the index file a pointer to the disk address of the index key record in the primary data file. To reduce on the number of disk accesses, multi level tree structured indexes such as ISAM, B-tree and B+-tree indexes are used. These multi level indexes are based on the binary search technique and have search time complexity of $O(\log_{fanout} bi)$ where fanout is the number of pointers that can be accommodated on each node of the tree. We focused in detail, on the construction of the B-tree and B+-tree and discussed the search, insert and delete operations on the B-tree and B+-tree. The differences between the B-tree and B+tree are discussed.

**

Part 2 of this Review Class: Final Exam Practice Discussions

The final exam practice posted to the class through blackboard is in the same format as the final exam with Section A (with 15 multiple choice questions) and Section B (with 4 5 questions). This year, I have reduced the mark for question 4 of Section B so that it has only 10 marks instead of previous 15

marks. Note that the final examination practice is provided so students actually tried to solve the questions and bring any problems they have for discussion. I usually do not post solution to practice tests except as a discussion in a review class. Thus, for those who solved the practice test, please, cross check your answers with the following. I will discuss only some questions in section B. Section A.

I also provide the instructions for the ONLINE FINAL EXAMINATION below before discussing the Final exam practice.
**

INSTRUCTIONS (Please Read Carefully)
Examination Period is 2 hours 30 minutes (online version time extended to 4 hours)
Answer all questions. Write your answers in the spaces provided in the question paper. This is closed book and closed notes test. You can type in your answers into the word file and submit, or print, write with hand, scan clearly into only a .pdf or .jpeg file and submit.
Total Marks =50. Total number of sections = 2
Please read questions carefully! Misinterpreting a question intentionally or unintentionally results in getting a "ZERO" for that question. Good Luck!!!

CONFIDENTIALITY AGREEMENT & STATEMENT OF HONESTY
I confirm that I will keep the content of this assignment/examination confidential.
I confirm that I have not received any unauthorized assistance in preparing for or doing examination. I confirm knowing that a mark of 0 may be assigned for copied work.
For Online Test/Examination in Comp 3150 Winter 2021: (additional rules to be observed):
1. I confirm that I agree to write this final examination as a closed book examination.
2. I confirm that I am the student with the name and student id signed below.
3. I confirm that I agree to not send email, chat, text or talk in any way to people other than the instructor or proctoring GA of this course during this examination.
4. I confirm that I agree to not engage in copying or cheating during this online examination.

_____ _____
Student Signature Student Name (please print)

_____ _____
Student I.D. Number Date

To write the Online Final examination, log on to Black board (BB) virtual class room at Exam time, and go to Resource page of the Black Board for Course, go to the Test Materials link to download the Course Comp 3150 Final Exam. You can do the test in one of these three ways:

1. Print it, write it with hand, scan clearly with camera saved only in a .pdf or .jpeg file and submit back through the assignment link before the end of test period.
2. Type in all your answers in the word file and submit.
3. Use a combination of methods 1 and 2 above.

Use of chat in BB is disallowed to reduce chances of cheating. Please, send mail to any of the GA or TAs or Professor Ezeife (cezeife@uwindsor.ca) if you have any questions. The GA/TA emails are given below:

For proctoring on Wed. April 21, 2021, arrive in the BB virtual class room at least 15 minutes before 08:30am. I will disable chat option for students to cut down on incidents of cheating. Students can

raise hands when they have a question so that the proctoring GA in charge of the student can communicate with them through email only. GA/TA will be responsible for students during proctoring as follows:

GA/TA Names	Email	Student to Proctor (Last name begins with)
Pooja Chakrabarthy	chakrabp@uwindsor.ca	A to C
Salwa Mohamed	moham12y@uwindsor.ca	D to J
Mohammad Elias Khan	khan1ft@uwindsor.ca	K to O
Niyati Vyas	vyas8@uwindsor.ca	P to R
Nihar Patel	patel3j@uwindsor.ca	S to Z

**
****

For marking purposes only (This part not to be filled by students)

Question	Mark
Section A (30 marks for 15 multiple choice questions)	/30
Section B	
Que 1 (20 marks)	/20
Que 2 (15 marks)	/15
Que 3 (20 marks)	/20
Que 4 (15 marks)	/15
Total	/100

***

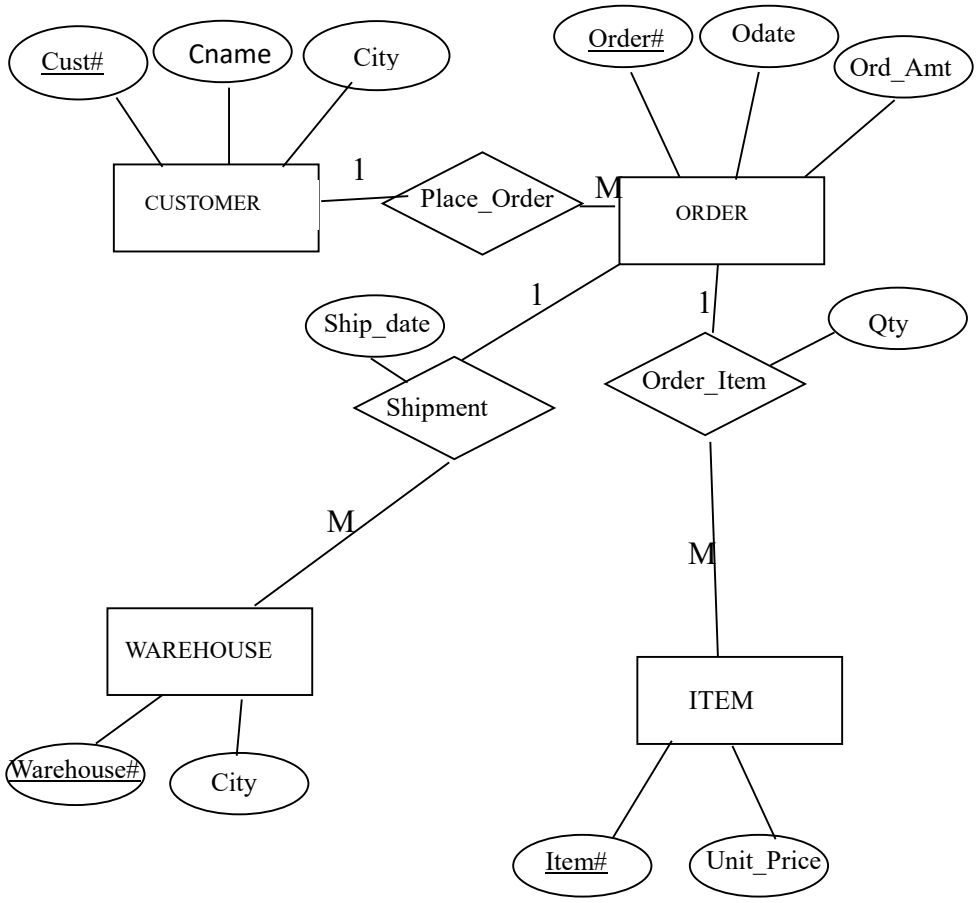
Solution (30 marks)

Question	Answer
1	a
2	a
3	d
4	b
5	b
6	d
7	d
8	d
9	e
10	a
11	b
12	d
13	a
14	a
15	d

Some Section B Solutions

Question 1. Here, note that the Cust# foreign key attribute of the CUSTOMER entity table had to be refined into an explicit relationship Place_Order so that the two entity tables CUSTOMER and ORDER can be connected through this explicit relationship table.

. 1 (20 marks for all of a to c)

Sub ques-tion	Answers
a.. (10 marks)	a. 
b.. (5 marks)	b. CREATE TABLE CUSTOMER (Cust# NUMBER(3) NOT NULL, Cname VARCHAR2(10), City VARCHAR2(10) , PRIMARY KEY (Cust#)); CREATE TABLE ORDER (Order# NUMBER(4) NOT NULL, Odate Date, ~~Cust# NUMBER(3),~~ Ord_Amt NUMBER(3), PRIMARY KEY (Order#)); CREATE TABLE ORDER_ITEM (Order# NUMBER(4) NOT NULL, Item# NUMBER(4) NOT NULL, Qty NUMBER(3), PRIMARY KEY (Order#, Item#), FOREIGN KEY (Order#) REFERENCES ORDER (Order#), FOREIGN KEY (Item#) REFERENCES ITEM(Item#)) ;

	CREATE TABLE ITEM (Item# NUMBER(3) NOT NULL, Unit_price NUMBER(5,2) , PRIMARY KEY (Item#)); CREATE TABLE SHIPMENT (Order# NUMBER(4) NOT NULL, Warehouse# NUMBER(3) NOT NULL, Ship_date DATE, PRIMARY KEY (Order#, Warehouse#), FOREIGN KEY (Order#) REFERENCES ORDER(Cust#), FOREIGN KEY (Warehouse#) REFERENCES SHIPMENT(Warehouse#); CREATE TABLE WAREHOUSE (Warehouse# NUMBER(3) NOT NULL, City VARCHAR2(10) , PRIMARY KEY (Warehouse#)); CREATE TABLE PLACE_ORDER(Cust# NUMBER(3) NOT NULL, Order# NUMBER(4) NOT NULL, PRIMARY KEY (Cust#, Order#), FOREIGN KEY (Cust#) REFERENCES CUSTOMER (Cust#), FOREIGN KEY (Order#) REFERENCES ORDER (Order#)) ;
c. (5 marks)	c. Get names of customers who ordered some items shipped from Windsor. Select Cname From CUSTOMER Where Cust# IN (Select Cust# From ORDER O, SHIPMENT S, WAREHOUSE W WHERE S.ORDER# = O.ORDER# AND S.WAREHOUSE# = W.WAREHOUSE# AND W.CITY = 'WINDSOR');

Question 2: (15 marks and in the new scheme will be 20 marks)

a. Design a simple database with at least 5 relations for such applications as "students taking courses taught by faculty in class rooms at a specific time" (5 marks)
b. Is the relation R in 3NF? Explain your answer using functional dependencies. (5 marks)
c. If your R above is in 3NF, create at least one table from your database that is not in 3NF. If your R is not in 3NF, decompose R into 3NF relations. In each case, you are required to specify the primary keys and the functional dependencies that exist in each of the relations and thus, in the database with some discussions.(5 marks).

Discussions: I have provided a lot of solutions in assignments 1, 2 and 3 as well as in midterms 1 and 2 for designing your own database.

For (a)
Hint again: is in the sentence given above, identify the entities such as students, courses, faculty, classroom so you can provide attributes for describing these entities such as:

Student (Sid, Sname, major, gpa)
Course(Cid, Ctitle, credithr)
Faculty(Fid, Fname, Salary)
Room(Rid, location, capacity)

However, you have only entity tables that are not linked together through any foreign key/primary key links in the above list. The relationship tables will define what actions tie each pair or more of these entities together and are part of the database. For example, we know that Students enrol in courses and this can be captured with the relationship enrol. Also, we know that in this mini world, faculty teaches courses in class rooms at specific time and thus, we have the two remaining following relationship tables to complete our database:

Enrol(Sid, Cid, Grade)
Teach(Fid, Cid, Rid, Time)

The functional dependencies (that is, business rules) existing and holding in this database are:

FD1: Sid -> {Sname, major, gpa}
FD2: Cid -> { Ctitle, credithr}
FD3: Fid -> {Fname, Salary}
FD4: Rid -> { location, capacity}
FD5: (Sid, Cid) -> Grade
FD6: (Fid, Cid, Rid) -> Time

For (b), From these FDs, we can see that the database is in 3NF because all non-key attributes are fully functionally and non-transitively dependent on the primary keys because there are no partial dependencies or transitive dependencies in any of the FDs in each of the tables.

*** Please, review your notes and all the previous solutions given of these types of questions and complete part c.

Question 3 Solution:

Que 3 (20 marks for all of a and b)

(a) Query in SQL, Relational Alge-bra, Relational Calculus, Result (10 marks)	(a). Retrieve the quantity and price of items ordered by a customer SQL: SELECT C.Cust#, C.Cname, R.Qty, O.Ord_Amt FROM CUSTOMER C, ORDER O, ORDER_ITEM r WHERE C.Cust#=O.Cust# AND O.Order# = R.Order# AND C.Cname = 'Peter'; Relation Algebra: $\pi_{Cust\#,Cname,Qty,Ord\_Amt}(\sigma_{Cname='Peter'}(CUSTOMER \bowtie_{cust.cust\#=order.cust\#} (ORDER) \bowtie_{Order.Order\#=order\_item.order\#} (ORDER\_ITEM)))$

	Relational Calculus: {c.cname, R.Qtyp, O.Ord_Amt | customer(c) and order_item(R) and order(O) and c.cust# = O.cust# and O.order# = R.order# and c.cname = 'Peter'} Query Result: Cust# Cname Qty Ord_Amt 1 Peter 1 100.25
(b) Query in SQL, Relational Algebra, Relational Calculus, Result (10 marks)	(b) Retrieve the city, order date and shipment date of an order. SQL: SELECT W.city, O.Odate, S.Ship_date FROM ORDER O, SHIPMENT S, WAREHOUSE W WHERE O.Order# = S.Order# AND S.WAREHOUSE# = W.WAREHOUSE# AND O.ORDER#=1123; Relational Algebra: $\pi_{City, Odate, Ship\_date}(\sigma_{Order.Order\#=1123}(ORDER) \bowtie_{Order.Order\#=Shipment.Order\#} (SHIPMENT) \bowtie_{Order.Order\#=order\_item.order\#} (WAREHOUSE)))$ Relational Calculus: {W.city, O.Odate, S.Ship_date | Warehouse (W) and Order (O) and Shiptment (S) and O.Order#=S.Order# and S.Warehouse# = W.Warehouse# and O.order# = 1123} Result: Windsor 25-May-16 10-Jun-16

Extra for Question 3: (c). Retrieve the average quantity of items ordered by each customer
SQL:

```
SELECT C.Cust#, C.Cname, avg(R.Qty)
FROM CUSTOMER C, ORDER O, ORDER_ITEM R
WHERE C.Cust#=O.Cust# AND O.Order# = R.Order#
Group by C.Cust#, C.Cname;
```

Relation Algebra:

$_{Cust\#,Cname}\Im_{avg\ Qty}$ (CUSTOMER $\bowtie_{cust.cust\#=order.cust\#}$ (ORDER) $\bowtie_{Order.Order\#=order\_item.order\#}$ (ORDER_ITEM))

Relational Calculus:

{c.Cust#, c.cname, avg(R.Qty) | customer(c) and order_item(R) and ∃(O) (order(O) and c.cust# = O.cust# and O.order# = R.order#)}

Query Result:

Cust#	Cname	avg(Qty)
1	Peter	1
2	Mary	2

Question 4 : (total marks = 15 for 7.5 + 7.5)

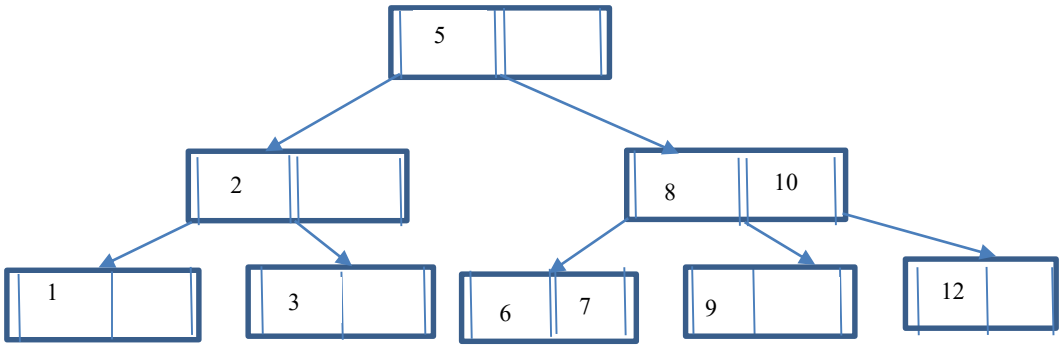
(a). Given the following unique id for some database records of a relation, construct the B-tree index structure of order p=3 (or with maximum of 3 pointers) for speeding up retrieval of these records from their primary data file stored on disk. The data to build index file for are:
8, 5, 1, 7, 3, 12, 9, 6, 2, 10
Show the steps through arriving at your final tree before giving your final tree.
(7.5 marks)

Note that this is a continuation of the B-tree of Fig 17.10 of the book I constructed and posted in my last lectures. The only difference here is that I asked to insert two more records of 2 and 10. Generally, for the final exam, we do not give a long chain of numbers to insert but ask to insert a few numbers (3 or 4) one at a time. Thus, it is important to study my few last lectures and may be the book to be able to insert a record into any B-tree or B+-tree and to be able to build on it.

To answer the question (a), above and add 2, then 10 in the B-tree of Fig 17.10, I got the final final B-tree below:

Solution (a) (7.5marks)

Final B-tree is given below:



This is the final B-tree after inserting 8, 5, 1, 7, 3, 12, 9, 6, 2, 10

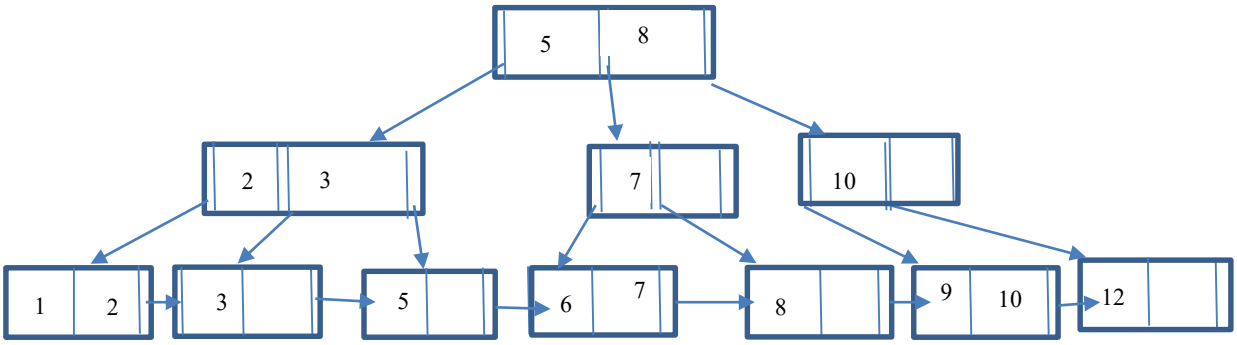
Steps for Getting the final B-tree are given below:

Show this on your own as I did in my classes for each step of building and adding each key to the B-tree

(b). Using the same data set of 8, 5, 1, 7, 3, 12, 9, 6, 2, 10, construct a B+-tree index structure of order p=3 (or with 3 [pointers for internal nodes and 2 data entries for leaf nodes).
Show the steps through arriving at your final tree before giving your final tree. (7.5 marks)

Solution (b) (7.5 marks)

Final B+-tree is given below:



This is the final B+-tree after inserting 8, 5, 1, 7, 3, 12, 9, 6, 2, 10

Steps for Getting the final B+-tree are given below:

Show this on your own as I did in my classes for each step of building and adding each key to the B+-tree.