1.
a)

```
GRANT CREATETAB TO A1;
```

b)

```
GRANT INSERT, DELETE ON EMPLOYEE, DEPARTMENT TO A2;
```

c)

```
GRANT UPDATE ON DEPARTMENT(DEPT_NAME) TO A3;
```

d)

```
REVOKE ALL FROM A4;
```

e)

```
CREATE VIEW A3_view AS
SELECT NAME, DOB, SALARY
FROM EMPLOYEE
WHERE DNO = 8;

GRANT SELECT ON A3_view TO A3 WITH GRANT OPTION;
```

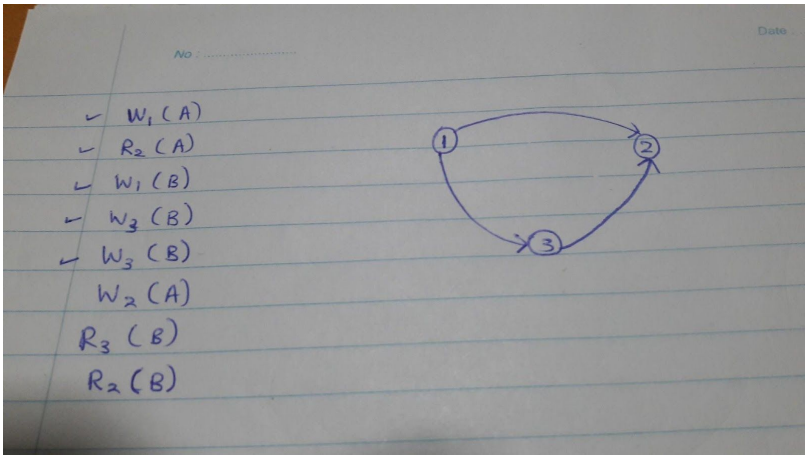
f)

```
CREATE TRIGGER total_salary
AFTER UPDATE ON EMPLOYEE
FOR EACH ROW
BEGIN
        IF new.Eid IS NOT NULL THEN
                UPDATE DEPARTMENT
                SET Total_sal = Total_sal - old.Salary
                WHERE Dno = old.Dno;
                SET Total_sal = Total_sal + new.Salary
                WHERE Dno = new.Dno;
        END IF;
END
```

2.

a)

i.

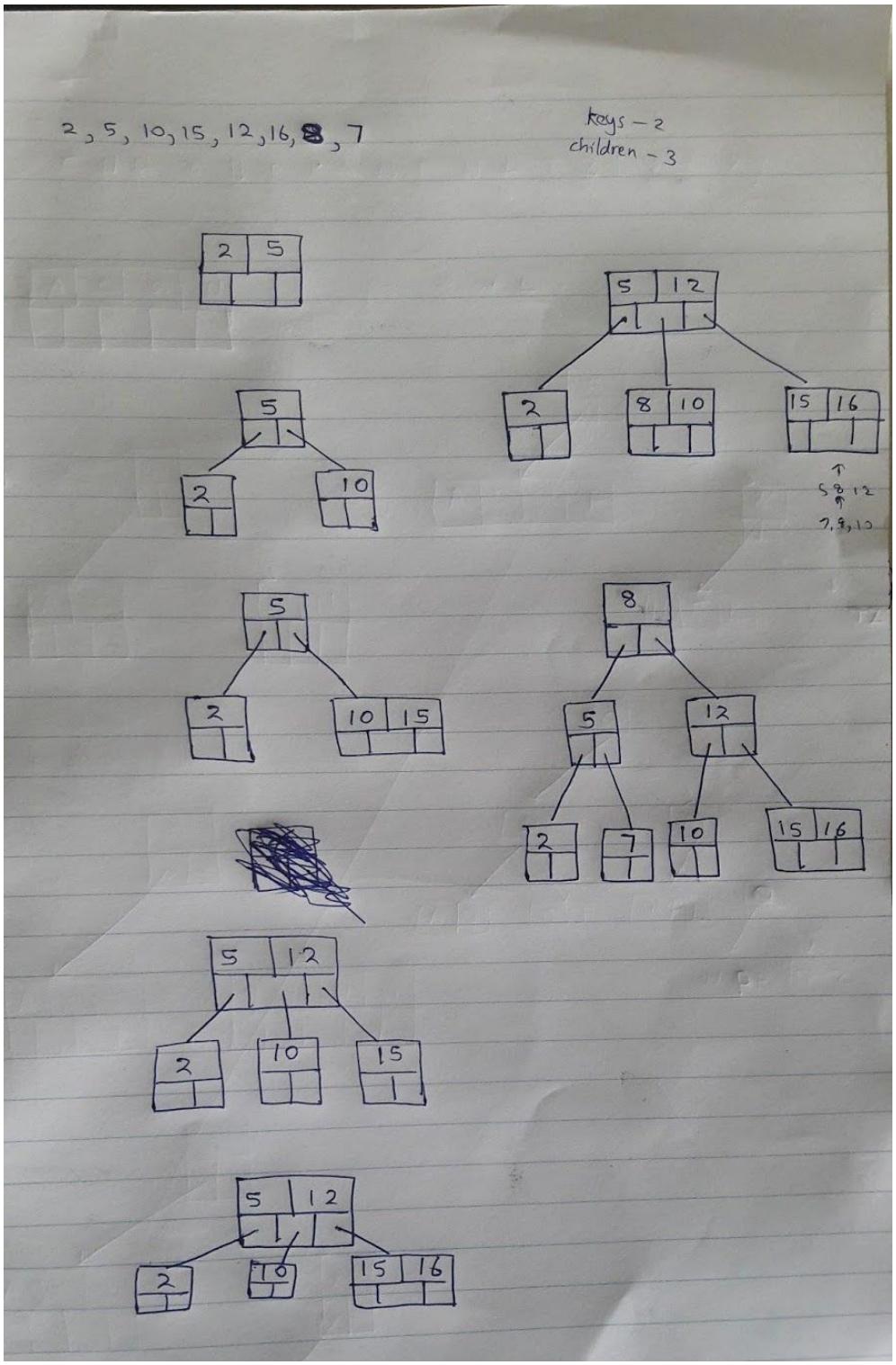


ii. yes conflict serializable because graph S has no cycles

iii. Since this is conflict serializable also should be view serializable

b)

c)



More: https://www.slideshare.net/kuldeep100/b-trees-dbms

3.
a)
Fname and Lname are set to be not null. So user have to enter data

b)
4th query does not run because Fname is given a null value.
8th query gives an error because ‘12113’ StudentID is inserted in the 3rd query.
Other entries execute and inserts the given data.

c)
DOB Date CHECK (DOB>1991-01-01)
or
ALTER TABLE Student
ADD CONSTRAINT check_date CHECK (DOB>1991-01-01);

d)
ALTER TABLE Student
ADD CONSTRAINT fk_StudentCourse
FOREIGN KEY CourseCode REFERENCES Course(CourseCode);

e)
i.

```
CREATE PROCEDURE count_emp_greater
BEGIN
        SELECT COUNT(OID) FROM occupation WHERE MIN_SALARY > 100000;
END

CREATE PROCEDURE count_emp_lesser
BEGIN
        SELECT COUNT(OID) FROM occupation WHERE MIN_SALARY < 100000;
END

CREATE PROCEDURE count_emp_equal
BEGIN
        SELECT COUNT(OID) FROM occupation WHERE MIN_SALARY = 100000;
END
```

ii.

```
call count_emp_greater();
call count_emp_lesser();
call count_emp_equal();
```