The University of New South Wales

COMP9315 DBMS Implementation Final Exam 14s2

Question 3 (14 marks)

Consider the following database tables:

```
create table Students (
        sid     integer,     -- 4 bytes
        name    varchar(30), -- average length 20 bytes (incl padding)
        bday    date,        -- 4 bytes
        degree  char(4),     -- 4 bytes
        primary key (id)
);
create table Courses (
        cid     integer,     -- 4 bytes
        code    char(8),     -- 8 bytes
        term    char(4),     -- 4 bytes
        title   varchar(40), -- average length 24 bytes (incl padding)
        primary key (cid)
);
create table Enrolments (
        course  integer,     -- 4 bytes
        student integer,     -- 4 bytes
        mark    integer,     -- 4 bytes
        grade   char(2),     -- 4 bytes (with padding)
        foreign key (course) references Courses(cid),
        foreign key (student) references Students(sid)
);
```

Now consider the following query on these tables:

```
select s.sid, s.name
from   Students s
         join Enrolments e on (s.sid = e.student)
         join Courses c on (c.cid = e.course)
where  c.code = 'COMP9315' and c.term = '14s2'
order  by name
```

A straightfoward translation of the above query into relational algebra would be:

```
Sort(
  Proj[sid,name](
    Sel[code=9315&term=14s2](
      Join[sid=student](Student, Join[cid=course](Courses,Enrolments))
    )
  )
)
```

and a simple mapping of this to an evaluation plan would be:

```
Tmp1 = Join[cid=course](Courses,Enrolments)
Tmp2 = Join[sid=student](Student,Tmp1)
Tmp3 = Sel[code=9315&term=14s2](Tmp2)
```

```
Tmp4 = Proj[sid,name](Tmp3)
Res  = Sort(Tmp4)
```

where the schemas for the intermediate tables are

```
Tmp1(cid,code,term,title,course,student,mark,grade)
Tmp2(sid,name,bday,degree,cid,code,term,title,course,student,mark,grade)
Tmp3(sid,name,bday,degree,cid,code,term,title,course,student,mark,grade)
Tmp4(sid,name)
Res(sid,name)
```

This question requires you to compute the costs of evaluating execution plans. In computing these costs, make the following assumptions:

- only nested-loop joins are available; 3 buffers are available for each join
- the result tuples from a join include all fields from both participating tables
- only external merge sort is available; 4 buffers are available for each sort
- projection in SQL does not eliminate duplicates (if there are any)
- the size of a tuple is simply the sum of the sizes of the fields
- to determine the size of a `varchar` field, simply use the average size
- all pages are 4KB and contain only tuples (no headers, etc.)
- the primary keys `cid` and `sid` each have a two-level B-tree index
- when indexes are used, their root page is locked in a memory buffer
- there are 5000 students, 1000 courses and 20000 enrolment records
- there are 70 students enrolled in COMP9315 14s2

If you think the above assumptions don't give enough information to answer this question, then state any additional assumptions that you are using.

Answer the following using the above scenario:

a. Compute the tuple size (*R*), page capacity (*c*) and number of pages (*b*)for each table.

b. Compute the cost for executing each operation in the above simple plan.
Compute all costs in terms of the number of pages read/written, **including** writing the final result.
(Hint: You will first need to compute number of tuples, tuple size, etc. for the result)

c. Give an execution plan that is likely to result in lower cost than the above plan.
You *do not* need to provide detailed cost analysis for the new plan,
but you should give a brief informal justification for why it is better.

Show all working.

Instructions:

- Type your answer to this question into the file called `q3.txt`
- Submit via: **`submit q3`**

End of Question