1. **Query Optimization** Given the following SQL query:

Student (sid, name, age, address)

Book(bid, title, author)

Checkout(sid, bid, date)

SELECT S.name

FROM Student S, Book B, Checkout C

WHERE S.sid = C.sid

AND B.bid = C.bid

AND B.author = ’Olden Fames’

AND S.age > 12

AND S.age < 20 And assuming:

* There are 10*,*000 Student records stored on 1*,*000 pages.
* There are 50*,*000 Book records stored on 5*,*000 pages. • There are 300*,*000 Checkout records stored on 15*,*000 pages.
* There are 500 different authors.
* Student ages range from 7 to 24.

1. Show a physical query plan for this query, assuming there are no indexes and data is not sortedon any attribute. **Solution:**

Note: many solutions are possible.

Scan:Book

Scan:Checkout

*1*

*sid*

Scan:Student

*1*

*bid*

*σ*

12

*<*

*age*

*<*

20

∧

*author*

=

0

*OldEnFames*

0

Π

*name*

Onthefly

Onthefly

Blocknestedloop

Tuple-basenestedloop

Figure 1: One possible query plan (all joins are nested-loop joins)

1. Compute the cost of this query plan and the cardinality of the result.

**Solution:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Cost | Cardinality | | Remarks |
| *S 1 C* | B(S) + B(S) \* B(C)  = 1000 + 1000 \* 15000  = 15001000 | 300000 join) | (foreign-key | (1) |
| (*S 1 C*) *1 B* | T(*S 1 C*) \* B(S)  = T(C) \* B(S)  = 300000 \* 5000  = 1500000000 | 300000 join) | (foreign-key | (2) |
| *σ* and Π | On the fly | 300000 \* *σauthor* \* *σage*  = 300000 \* 5001 ∗ 187  ≈ 234 | | (3) |
| Total | 1515001000 | 234 | |  |

* 1. We are doing page at a time nested loop join. Also, the output is pipelined to next join.
  2. The output relation is pipelined from below. Thus, we don’t need the scanning term for outer relation.
  3. We assume uniform value distributions for age and author. We assume independence among participating columns.