PRINCIPLES OF DATA ORGANISATION

Non-spatial join

MOTIVATION

- Key, pointer pairs ~ index.
- Unlike hashing, trees allow retrieving a set of records with keys from a given range
- Join on multiple query conditions
- For simplicity we focus only on equi-joins (the join predicate is equality)

NESTED LOOP JOIN

- Nested loop join checks one by one for each element of a dataset *R* all elements in dataset in *S*
- Traditional join in relational databases (relationaj join – we join relations)
- In its basic version, the nested loop join is the least efficient algorithm from the datasetal joins algorithms
  - The condition can be any, not just equality

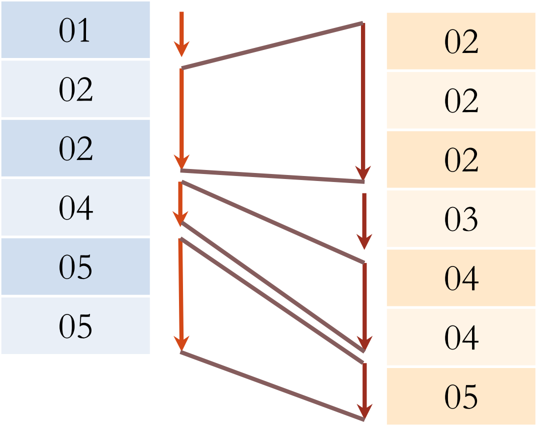
```
FOREACH r ∈ R DO
  FOREACH s ∈ S DO
    IF cond(r,s) THEN
      REPORT(r,s)
```

SORT-MERGE JOIN

- Two-phase algorithm:

1. Sort both datasets R, S independently
2. Scan both datasets at once in the same order and join

- We do not mutually compare everything
- We have to sort the data
  - Or we may get them sorted



HASH JOIN

- One of the datasets (R) is hashed with a hash function $\boldsymbol{h}$
- The other dataset (S) is processed one by one and the elements' ids are hashed with $\boldsymbol{h}$
- We get a bucket of candidates to check for the predicate
- If for two elements $r \in R$, $s \in S$: $h(r) = h(s)$, then r and s are checked for $r.cmpr\_attributes = s.cmpr\_attributes$