Exercise 1a — TPMMS, Change Two Parameters

Suppose you run **Two-Phase Multiway Merge Sort** on a file as in the lecture.
Tell how many I/O's are needed for the sort if:

a) The file size (number of records) is tripled.
b) Main memory size (number of available buffers) is doubled, but file size stays the same.

Assume the algorithm still finishes in exactly **two phases**.

Exercise 2a — TPMMS with Different File Parameters

You want to use **Two-Phase Multiway Merge Sort** (or more phases if needed) to sort a file.
The file contains **240,500 records**.
Each block can store **25 records**.
You have **12 memory blocks**.

a) How many blocks are needed to store this file?
b) How many sorted runs (sublists) result after the first phase?
c) Can the file be sorted in 2 phases? If no, how many phases are necessary?
d) How many I/O operations are required in total?

Exercise 3a — Set & Join Operations on Given Relations

Consider the relations:

```
R(A,B):
A  B
x 1
y 2
z 3
y 4

S(B,C):
B  C
1 5
4 6
2 7
3 8
```

Compute the following:

a) R ⋈ S
b) R ⋈A=C S (assume attribute A in R joins with B in S)
c) πA(R ⋈ S)
d) πB(R) ⋈ S

---

1) You want to use Two-Phase Multiway Merge Sort to sort a file.

The file contains **360,000 records**.
Each block can store **30 records**.
There are **15 main memory blocks** available.

a) How many blocks does the file occupy?
b) How many sorted runs (sublists) exist after Phase 1?
c) Is 2-phase merging possible with 15 blocks? If yes, how many runs are merged at once?
d) What is the total I/O cost of the complete sort?

---

2) Assume you use Two-Phase Multiway Merge Sort, but the available main memory is reduced.

A file contains **250,000 records** and each block stores **25 records**.
You only have **8 blocks of main memory**.

a) How many blocks are required to store the file?
b) How many initial runs result after the first sorting phase?
c) Can TPMMS (2-phase) still be applied? If not, how many phases are needed?
d) What is the total number of I/O operations for sorting?

Exercise Sheet 2, which includes:
1. Two-phase multiway merge sort cost analysis
2. Sorting file parameters (records, blocks, memory, I/Os)
3. Relational algebra computation (joins and projections).

Here are **two brand-new exercise questions** written **in the same structure, format, and level** as those on your sheet — ideal for continuation (they could be numbered *Exercise 4* and *Exercise 5*).

◆ Exercise 1a — Two-Phase Multiway Merge Sort (Extended)

You want to sort a file using the **Two-Phase Multiway Merge Sort** algorithm.
The file consists of **240,000 records**.
Each block can hold **30 records**, and there are **15 main-memory blocks** available.

a) How many blocks are needed to store the file if every block is filled completely?
b) How many sorted sublists (runs) are created after Phase 1?
c) How many runs can be merged at once in Phase 2?
d) Will two phases be sufficient to complete the sort?
e) Calculate the total number of I/Os (reads + writes) for the full sort.
f) How would the cost change if the number of available buffers increased to 30?

🔹 Exercise 2a — Sorting and Joining Relations

You need to sort and then join two relations **R(A,B)** and **S(B,C)** using a **Sort-Merge Join**.

Parameter	R	S
Number of records	80,000	40,000
Record size	200 bytes	300 bytes
Block size	4000 bytes	4000 bytes
Main-memory buffers	20	—

a) Compute how many blocks are needed to store each relation.
b) Determine the number of initial sorted runs for R and S (Phase 1).
c) Check if two phases are sufficient to sort both relations; if not, how many phases are required?
d) Estimate the total I/O cost (reads + writes) to sort R and to sort S.
e) Assuming both are sorted, estimate the I/O cost of the **Sort-Merge Join**.
f) If block size doubles to 8192 bytes, how does this affect the number of runs and the total I/O cost?

✅ These mirror the structure of your original **Exercise 1 & 2**:

- Same step-by-step subparts (a–f).
- Cover **block calculations, runs, phases, and I/O cost**.
- Exercise 5 extends naturally to a **join** scenario (like advanced lecture problems).

Would you like me to provide **worked solutions** for Exercise 4 or 5 (showing the full numeric reasoning for runs, phases, and I/Os)?

🔹 Exercise 4 — External Sorting and Join Combination

You want to sort and then join two relations **R** and **S** on attribute **B**.
The sorting uses the **Two-Phase Multiway Merge Sort** algorithm.
The join uses a **sort-merge join** once both relations are sorted on **B**.

Given Data:

Parameter	Relation R	Relation S
#Records (T)	60,000	20,000
Record size	200 bytes	250 bytes
Block size	4000 bytes	4000 bytes
Records per block	20	16
Main memory blocks available	15	—

Tasks

a) Compute how many **blocks** are needed to store each relation if blocks are completely filled.

b) Determine how many **initial sorted runs (sublists)** are produced for each relation in **Phase 1** of the two-phase multiway merge sort.

c) Verify whether both relations can be fully sorted in **two phases** with 15 buffers.

- If not, explain how many phases would be needed.

d) Estimate the **I/O cost** (reads + writes) for sorting **R** and **S** individually.

e) Assuming both are sorted, estimate the **I/O cost** for a **sort-merge join** between **R** and **S**.

f) What would change in the total I/O cost if you used **block size = 8192 bytes** instead?

✅ **This new exercise mirrors your Sheet 02**:

- Same structure (given parameters → stepwise subquestions).
- Combines both **external sort** and **join cost estimation** — a realistic DBMS exercise.
- Encourages showing intermediate reasoning (blocks, runs, phases, I/Os).

Exercise A — TPMMS (Two-Phase Multiway Merge Sort)

You use Two-Phase Multiway Merge Sort with 2 phases.

- File has **240,000 records**
- Block size = **4096 bytes**, block header = **96 bytes**
- Record size = **128 bytes**
- Main memory can hold **12 blocks**

a) How many blocks does the file occupy?
b) How many sorted chunks after phase 1?
c) How many phases are required (2 phases or more)?
d) Total I/O cost of the full sorting?

Exercise B — TPMMS Under Variation

Take the setup of Exercise A and answer these variants:

1. Memory is doubled — now **24 blocks fit in RAM**.
   → How does this change the #chunks and I/O?
2. Records increase by factor **3** (now 720,000 records) but memory stays 12 blocks.
   → Recompute blocks, chunks, I/Os.

Exercise C — Join and Projection with IO cost

You have relation **T(X,Y,Z)** with:

- 120,000 tuples
- Record size = 200 bytes (+40 bytes header)
- Block size = 8192 bytes (+160 bytes header)
- Attributes X,Y,Z are 4B, 50B, 120B

We want to compute:

$$\pi_Y(\sigma_{X=50}(T))$$

where 8% of tuples satisfy X=50, and projection eliminates duplicates — on average each distinct Y appears twice.

Compute:
a) #tuples after selection
b) #tuples after projection
c) #blocks needed to read original T
d) Cost in IOs if we scan T, apply both ops in memory, and write only the final result

---