Question 1 [25] Answer the questions below

i. (5 points) Reading pages from disk to main memory constitutes majority of the total cost in a DBMS in many traditional database applications. Briefly describe in 1-3 sentences why this is the case.

ii. (20 points) Two strategies to reduce disk access cost are:

1. Placing the database on the disk by cylinders.
2. Splitting the database to multiple disks.

TPMMS (Two Phase Multiway Merge Sort) Algorithm is used to sort a database when the size of that database is larger than the main memory. For the “read” and “write” steps of each phase of the TPMMS algorithm do the strategies (1) and (2) above help improving the disk access cost. Explain your reasoning for each briefly in 1-3 sentences.

Question 2 [25] Make the following assumptions.

1. A bucket can hold two keys and a pointer.

2. Database D initially contains only one object with key 10100.

3. Six objects with the following keys are inserted to D in the following order: 00110, 11010, 10011, 01010 and 10110.

Answer the questions below based on these assumptions

i. (10 points) Assume that extensible hash table is used to index the database. Show the index structure after insertion of each of the six objects. Use the bits starting from left to right while building the index. This is the strategy used in the text book.
ii. (15 points) Assume that linear hash table is used to index the database with the restriction that at most 80% of the hash table can be full at any time. Show the index structure after insertion of each of the six objects. Use the bits starting from right to left while building the index. This is the strategy used in the text book.

Question 3 [20] Answer the questions below:

i. (10 points) What is the difference between primary and secondary index? Describe one advantage of secondary index over primary index.

i. (5 points) Describe an advantage of B-tree over linear hash table.

ii. (5 points) Describe a scenario when linear hash table should be preferred over extensible hash. Briefly explain your reasoning.

Question 4 [30] Assume that we have a database of three dimensional objects. The location of each object is represented by its (x, y, z) coordinates. Assume that our DBMS supports only B-tree as an index structure. However, it allows us to build as many B-trees as we like.

Two fundamental queries on this database are range and nearest neighbor queries. These queries are defined as follows

- **Range query:** Find the objects whose coordinates are within a given interval. In other words, find objects \( r \) that satisfy

\[
(x_{min} \leq r.x \leq x_{max}) \ \text{AND} \ (y_{min} \leq r.y \leq y_{max}) \ \text{AND} \ (z_{min} \leq r.z \leq z_{max}).
\]

- **Nearest neighbor query:** Find the object in the database that is closest to a given query object \( q \). The distance between query \( q \) and a database record \( r \) is defined as the infinity norm:

\[
dist(q, r) = \max\{|q.x - r.x|, |q.y - r.y|, |q.z - r.z|\}
\]

Based on these constraints, answer the following questions.

i. (15 points) Describe an algorithm to answer a range query on this database efficiently using three B-trees (i.e., one B-tree on each dimension).

ii. (15 points) Describe an algorithm to answer a nearest neighbor query on this database efficiently using three B-trees (i.e., one B-tree on each dimension).