1. (20 points) Given a B+tree index on composite search key \( \langle \text{age}, \text{grade} \rangle \), explain whether it is possible to evaluate the following query with an index-only plan:

\[
SELECT S\.age, \text{MIN (grade)} \\
FROM Student S \\
WHERE S\.age >= 30 AND gender = 'Female' \\
GROUP BY S\.age
\]

2. (40 points) Consider the B+ tree index shown in Figure 1, which stores data records as its data entries in the leaves. Each intermediate node can hold up to five pointers and four key values. Each leaf can hold up to four records, and leaf nodes are doubly linked as usual, although these links are not shown in the figure. Answer the following questions:

(a) Name all the tree nodes that must be fetched to answer the following query: “Get all records with search key less than 82.”

(b) Show the B+ tree that would result from inserting a record with search key 92 into the tree.

(c) Show the B+ tree that would result from deleting the record with search key 43 from the original tree.

(d) Name a search key value such that inserting it into the (original) tree would cause an increase in the height of the tree.

3. (40 pts) Answer the following questions about extendible hash indexes. Use your answer from part (1) to answer parts (2), (3), and (4).

(a) Show an Extendible Hashing index with data entries \{3, 5, 7, 9, 10, 15, 25, 31, 44, 64\} using 4 buckets, where each bucket can hold 4 data entries.

(b) What is the maximum number of data entries that can be inserted (given the best possible distribution of keys) before you have to split a bucket? Explain briefly.

(c) Show the index after inserting a single record whose insertion causes a bucket split.

(d) What is the minimum number of record insertions that will cause a split of all four buckets? Explain briefly.
Figure 1: Sample B+-tree