Question-1: Let relations r_1(A, B,C) and r_2(C, D, E) have the following properties: r_1 has 20,000 tuples, r_2 has 45,000 tuples, 25 tuples of r_1 fit on one block, and 30 tuples of r_2 fit on one block. Estimate the number of block transfers and seeks required, using each of the following join strategies for ($r_1 \bowtie r_2$):

a) Hash join.
b) Block nested-loop join.

Question-2: Suppose you need to sort a relation of 40 gigabytes, with 4 kilobytes block, using a memory size of 40 megabytes. Suppose the cost of a seek is 5 milliseconds, while the disk transfer rate is 40 megabytes per second.

a) Find the cost of sorting the relation, in seconds, with the number of buffers allocated are 2, i.e., b_b=2.
b) How many merge passes are required in case of b_b=100?

Question-3: Consider the relations r_1(A, B,C), r_2(C, D, E), and r_3(E, F), with primary keys A, C, and E respectively. Assume that r_1 has 1000 tuples, r_2 has 1500 tuples and r_3 has 750 tuples. Estimate the size of $r_1 \bowtie r_2 \bowtie r_3$, and give an efficient strategy for computing the join.

Question-4: Suppose that a B+ tree index on (dept_name, building) is available on relation department. What should be the best way to handle the following selection?

$\sigma_{\text{(building < “Watson”)}} \Lambda_{\text{ (budget < 55000)}} \Lambda_{\text{ (dept_name = “music”)}}$(department)