1. Suppose that you have an application in which you want to use B-trees. Suppose that the computer you will be using has disk blocks holding 4096 bytes, the key is 4 bytes long, each child pointer (which is a disk block id) is 4 bytes, the parent is 4 bytes long and the data record reference (which is a disk block id along with a offset within the block) is 8 bytes.

You have an application in which you want to store 1,000,000 items in your B-tree. What value would you select for \( t \)? (Show how you derived it.) What is the maximum number of disk pages that will be brought into main memory during a search? Remember that the root is kept in main memory at all times.

2. Which of the following are legal B-trees for when the minimum branching factor \( t = 3 \)? For those that are not legal, give one or two sentence very clearly explaining what property was violated.

   i)   ii)   iii)   iv)   v)
   FV   D   CF
   KS   ABC   AB
   WXY   FGH   D
   GH   LMO   GHIJ
   TU
   BD
   GJ
   H
   J

3. Show the B-tree that results when inserting R,Y,F,X,A,M,C,D,E,T,H,V,L,W,G (in that order) branching factor of \( t = 3 \). You need only draw the trees just before and after each split.

4. Show the B-tree the results when deleting A, then deleting V and then deleting P from the following B-tree with a minimum branching factor of \( t = 2 \).

5. Show a way to represent the original B-tree from problem 4 as a red-black tree. You can indicate the color of each node by circling it with red or black or just by putting a “r” or “b” next to it.