Problem

Prove that if A is a language in L, a family of branching programs (B_1,B_2, . . .) exists wherein each B_n accepts exactly the strings in A of length n and is bounded in size by a polynomial in n.

Step-by-step solution

Step 1 of 2

A branching program is defined as "**a directed acyclic graph** where the variables are used to label all the nodes except only two output nodes which is labeled 0 and 1". The **query nodes** are defined as all the nodes which are labeled by the variables. All the query nodes consists two outgoing edges, labeled as 0 and 1. Both output nodes doesn't consists outgoing edges.

Comment

Step 2 of 2

Consider a language A which takes an **input length** of n. A set of **branching programs** is taken in such a way that each branching program accepts exactly the strings in A of length.

• Now, the **Merge-sort** can be implemented as a circuit in which the input length of language A has taken as the nodes of the branching program.

• It is used to compare two bits after recursively dividing the given inputs in to half. The total time taken here (to divide the inputs into equal halves iteratively) is $\log n$.

• Consider the inputs can be called as x_1, x_2 and the outputs can be called as y_1. Now, **the action of the merge-sort algorithm** can be mimicked on an array. It can be implemented one step at position to be the n input, $n/2$**-output sub-circuit**.

• Now, **a pass** can be implemented as the serial concatenation of steps, which has a size $n \log n$. Therefore, this gives a size $n * \log n = O(n \log n)$.

• Therefore, it can be said that "a language with an input length of n can be computed in $O(n \log n)$ size circuits by using branching program.

Hence from the above explanation it can be said that the language A is in logarithmic space. In other words, the language the language A is in L.

Comment