Maman 11 - Semester 05b

Question 3. For all $m \geq 0$ and $n \geq m$, the result of replacing each of the nodes of the binomial tree B_m by the binomial tree B_{n-m} (where the original children of the replaced node become the children of the root of the replacing tree) is the binomial tree B_n.

Answer. We start by restating the question in a recursive manner, afterwards we will show that the two definitions are equivalent.

Question 3 restated. For all $m \geq 0$ and $n \geq m$, the result of invoking the following algorithm A on B_m and $(n-m)$ is the binomial tree B_n.
$A(B_m, n-m)$:

1. Decompose B_m into the root and the list of its children: $B_{m-1}, \ldots, B_0$.
2. For each $0 \leq k \leq m-1$ invoke $A(B_k, n-m)$, let's call the result list of "trees" [1] $B_{m-1}^{n-m}, \ldots, B_0^{n-m}$.
3. Create the binomial trees $B_{m-1}, \ldots, B_0$.
4. Compose a new tree from a new root r whose ordered children are $B_{m-1}^{n-m}, \ldots, B_0^{n-m}, B_{m-1}, \ldots, B_0$.

The actions of algorithm A are best demonstrated by Figure 1.

We prove that the result of $A(B_m, n-m)$ is B_n by induction on m.

1. $m = 0$.

   Hence the single node is "replaced" by the tree $B_{n-0} = B_n$ and thus the resulting tree is B_n.

2. Assuming for m we will prove for $m+1$.

   By the induction hypothesis the "trees" $B_{m-1}^{n-m}, \ldots, B_0^{n-m}$ are the binomial trees $B_{n-1}, \ldots, B_{n-2}, \ldots, B_{n-m-1}$. Hence the children of r are $B_{n-1}, \ldots, B_0$ and thus the result is the binomial tree B_n.

[1] Currently we do not know that they are indeed trees.

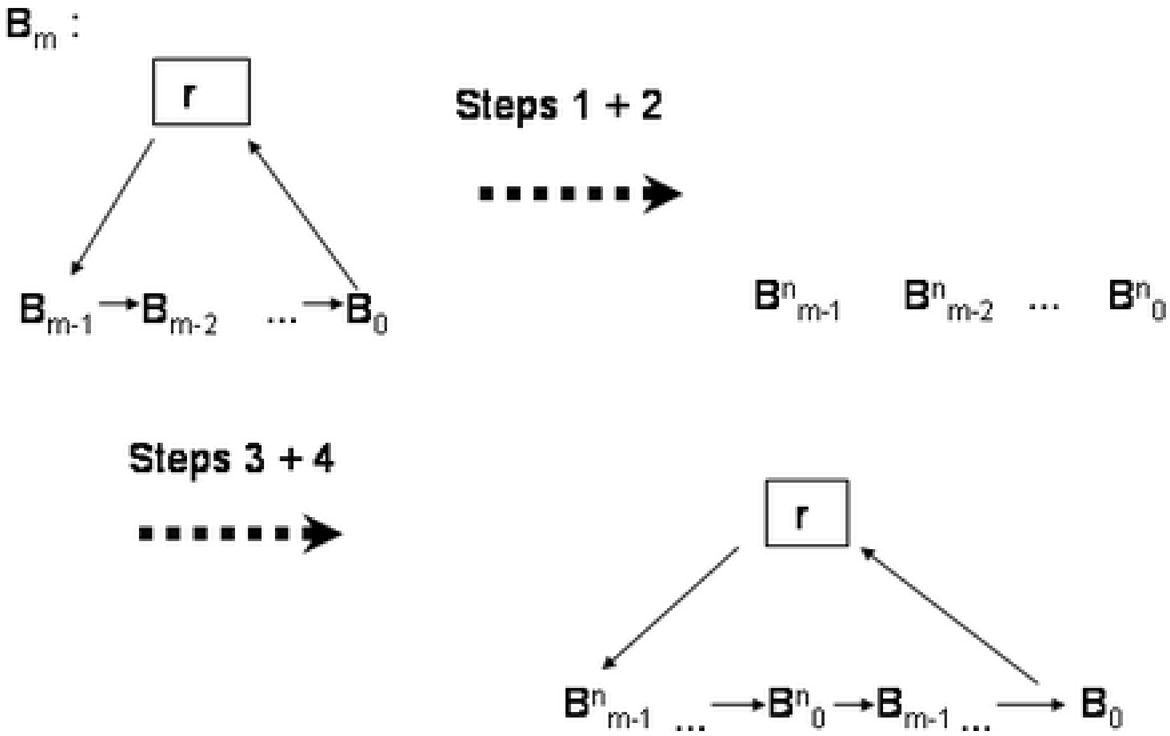

Figure 1: Algorithm A with inputs $(B_m, n - m)$.

Proving the equivalence between the two form of question 3. In the way question 3 is stated in the Maman, the exact way the original children of the node are connected to the children of the replacing root is not defined. Here we assume that the children of the node are linked **before** the children of the replacing tree. Note that the above assumption is a must as otherwise the claim is not true (check for example what happen where $n = m = 1$ and the child of the root of B_m is not linked in the order mentioned above).

The equivalence between the original form of question 3 and the restated form is proved using the following observations about the way the replacements are done in the original question.

Remark 1 *Replacing a node only effects the structure of the subtree rooted at it.*

Proof: Let's assume that in the replacement operation the node itself is not replaced, but rather the node stays in place and the children of the replacing tree are added after the children of the node. Note the final effect is not changed and therefore we can assume w.l.o.g. that this is indeed the way replacements are done. Having the new form of replacements it is clear that replacing a node does not have any effect on the parts of the tree that are not its descendants. ∎

Remark 2 *The order of the replacements does not effect the final result.*

Proof: The proof can be easily done using induction. However since it is not hard and it is not the main issue of the question we allow ourself to omit it. ∎

Claim 3 *The two forms of the question are equivalent.*

Proof: By Remark 2, we may assume w.l.o.g. that the replacements are done in a bottom up order (i.e., from the leaves up). By Remark 1 we know that the above operation is the same as doing the replacements for each of the subtrees of the root independently and then doing the replacement to the root itself. Hence it is the same as invoking the replacing algorithm for each of the subtrees following by replacing the root. Therefore the operations being taken are completely the same as in the restated question. ∎