2. Only those positions are combined which have different characters to the left. Thus it guarantees left-maximality of the repeats. Recall that we consider processing the edge $v \to w$ and let a be the first character of the label of this edge. The annotation already computed for x was inherited along edges outgoing from v, that are different from $v \to w$. Thus the first character of the label of such an edge, say σ, is different from a. Now since ν is the repeated substring, σ and a are characters to the right of ν. As a consequence, only those positions are combined which have different characters to the right. In other words, the algorithm also guarantees right-maximality of the repeats.

As soon as for the current edge the repeats are reported, the algorithm computes the union $A(v, \sigma) \cup A(w, \sigma)$ for all characters σ, i.e. the annotation is inherited from node w to node v. In this way, after processing all edges outgoing from v, this node is annotated by the set of positions where ν occurs, and this set is divided into (possibly empty) disjoint subsets $A(v, \sigma_1), \ldots, A(v, \sigma_r)$, where $\Sigma = \{\sigma_1, \ldots, \sigma_r\}$.

Example 9.17 The suffix tree of the string $s = \mathtt{x}ggcgc\mathtt{y}gcgcc\mathtt{z}$ is shown in Figure 9.8. We assume the leaf annotation of it (as shown in Figure 9.9) is already determined.

Proceeding from leaves 8 and 3, the bottom up traversal of the suffix tree for $\mathtt{x}ggcgc\mathtt{y}gcgcc\mathtt{z}$ begins with node v whose path-label is $gcgc$ of depth $q = 4$. This node has two children which do not have the same character to their left ($\mathtt{y}$ vs. g). The node is annotated with $(g, 3)$ and $(\mathtt{y}, 8)$. Because our repeat is also left-maximal, it is reported: $((3, 3 + 4 - 1), (8, 8 + 4 - 1)) = ((3, 6), (8, 11))$, as can be seen in Figure 9.10, which shows the annotation for a large part of the suffix tree and some repeats.
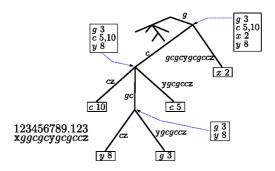

Figure 9.10: The annotation for a large part of the suffix tree of Figure 9.9 and some repeats.

Next comes the node v with path-label gc of depth two. The algorithm starts by processing the first edge outgoing from v. Since initially there is no annotation for v, no repeat is reported and v is annotated with the label of leaf 10: $(c, 10)$. Then the second edge is processed. This means that the annotation $(g, 3)$, $(\mathtt{y}, 8)$ for w with path-label $gcgc$ is combined with the annotation $(c, 10)$. The new annotation for v becomes $(c, 10)$, $(\mathtt{y}, 8)$, $(g, 3)$. This gives the repeats $((8, 9), (10, 11))$ and $((3, 4), (10, 11))$. Finally, the third edge is processed. $(c, 10)$ and $(c, 5)$ cannot be combined, see condition 2 above. So only the repeats $((5, 6), (8, 9))$ and $((3, 4), (5, 6))$ are reported, resulting from the combination of $(\mathtt{y}, 8)$ and $(g, 3)$ with $(c, 5)$. The final annotation for v is $(c, 10)$, $(\mathtt{y}, 8)$, $(g, 3)$, $(c, 5)$, which can also be read as $A(v, g) = \{3\}$, $A(v, c) = \{5, 10\}$ and $A(v, \mathtt{y}) = \{8\}$. We leave further processing up to the reader. ■