We defined the product of two elements. The product of a and b is ab. We now want to define the product of n elements and prove that the usual exponentiation rules are valid. The rest of this paragraph is extremely dull. The reader may just glance at the assertions and skip the proofs if she (or he) wishes.

By the product of three elements a, b, c in a group G, we understand an element abc of G. Let us recall we agreed to denote by abc the element $(ab)c = a(bc)$. So the product of a, b, c in this order is evaluated by two successive multiplications. Either we evaluate ab first, then multiply it by c, or we evaluate bc first, then multiply a by it. In either way, we get the same result by associativity and this result is denoted by abc, without parantheses.

Now let us consider the product of four elements a, b, c, d. Their product in this order will be defined by three successive multiplications of two elements. This can be done in five distinct ways:

$$a(b(cd)), a((bc)d), (ab)(cd), ((ab)c)d, (a(bc))d,$$

but these five products are all equal by associativity. The first two products are equal since $(ab)c = a(bc)$. Further, we have $a(b(cd)) = (ab)(cd)$ [put $cd = e$, then $a(be) = (ab)e$] and $(ab)(cd) = ((ab)c)d$ [put $ab = f$, then $f(cd) = (fc)d$]. So the five products are equal. This renders it possible to drop the parentheses and write simply $abcd$. This is the product of a, b, c, d in the given order.

More generally, we want to define the product of n elements $a_1, a_2, ..., a_n$ in a group $G(n > 2)$. The product of $a_1, a_2, ..., a_n$ will be defined by $n-1$ successive multiplications of two elements. By inserting parantheses in all possible ways, we obtain many products (their exact number is 2, 4...$(4n-6/n!)$, but associativity assures that these products are equal. Now we prove this. In view of some later applications, the following lemma is stated more generally than for groups.

Lemma 0.1. [1] *Let G be a nonempty set and let there be defined an associative binary operation on G, denoted by juxtaposition. Let*

[1] 8.3