

Задача 2. Задачата да се реши на един от езиките C, C++ или Java. В началото на решението си посочете кой език сте избрали.

Разглеждаме кореново дърво, във възлите на което има записани двойки от символ (**char**) и цяло число (**int**). Всеки възел на дървото може да има произволен, краен брой наследници. За удобство разглеждаме функциите *sym* и *val*, дефинирани над множеството от възлите на дървото, така че за всеки възел *v* на дървото, в който е записана двойката $\langle a, b \rangle$, $sym(v) = a$ и $val(v) = b$.

Клон в дървото *T* ще наричаме всеки път $\pi = (v_0, v_1, \dots, v_n)$, за който v_0, \dots, v_n са върхове на *T*, v_n е листо на *T*, а v_i е родител на v_{i+1} за всяко $i < n$.

За всеки клон $\pi = (v_0, v_1, \dots, v_n)$ на дървото *T* дефинираме съответни “дума” и “стойност” по следния начин:

$word(\pi) = sym(v_0)sym(v_1)\dots sym(v_n)$,

$$value(\pi) = \sum_{i=0}^n val(v_i),$$

т.е. $word(\pi)$ е думата, която се получава от последователното прочитане на символите, записани във възлите на пътя, а $value(\pi)$ е сумата на числата, записани в тях.

а) Да се избере, дефинира и опише подходящо представяне на дърво от описания тип.

б) За така дефинираното представяне да се реализира функцията:

int sumVal ([подходящ тип] T, [подходящ тип] u, [подходящ тип] v),

която по дадено дърво *T* и два негови върха *u* и *v* намира и връща сумата от всички стойности $value(\pi_u) + value(\pi_v)$, за които π_u и π_v са клони с начала *u* и *v* (съответно) със свойството $word(\pi_u) = word(\pi_v)$. Ако такива клони няма, сумата се счита за 0.

Забележки:

1. В зависимост от избраното представяне, параметърът *T* може да бъде пропуснат.
2. Не е нужно дефиницията на представянето на дървото да бъде пълна, нужно е само да е достатъчна за реализацията на функцията **sumVal**.
3. Позволено е използването на функциите и класовете от стандартната библиотека на избора от Вас език.