

Задача 1. Да се напише функция, която намира **произведението** на всички елементи от двоично наредено дърво от цели числа, които имат **четен брой** наследници в интервала $[x, y]$ (т.е. сумата на броя на елементите в лявото поддърво и броя на елементите в дясното поддърво, които са в интервала $[x, y]$, е четно число). Нека x и y са параметри на функцията.

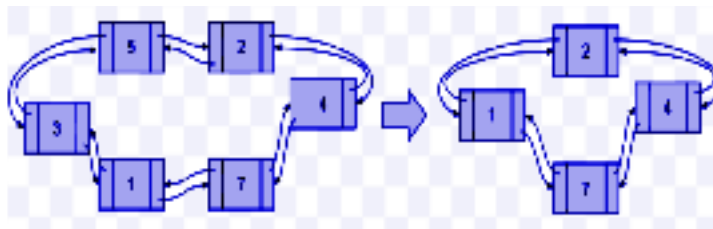
При обхождане на дървото да се използва наредеността на дървото, където е удачно.

Задача 2. Дефиниран е шаблон на двусвързан цикличен списък (без фиктивен елемент). Напишете функция, която получава като аргумент указател към един негов елемент и премахва от него всички елементи, които са равни на сумата от двата си съседа.

След изпълнение на функцията списъкът трябва да остане

валиден и да не съдържа елементи с това свойство. .

```
// Шаблон на елемент на
// двусвързан цикличен списък
template <typename T>
struct cir_list_elem {
    T data;
    elem_cir<T> *next;
    elem_cir<T> *prev;
};
```



Пример за специализация на двусвързания цикличен списък с цели числа

Премахва се 5 - сума от елементите 2 и 3;

Премахва се 3 - сума от елементите 1 и 2;

Задача 3. Нека имаме мрежа от колектори за вода. Всеки колектор се идентифицира от уникално число. За всеки два колектора знаем дали има тръбопровод между тях, посоката, в която върви водата между тях и **процентът** вода, който би се загубил при преноса на водата между тези два колектора. Освен това знаем, че съществуват опасни колектори, до които не искаме да достига водата, тъй като там ще изчезне безследно. Да се реализира функция, която по подадени колектори **a** и **b**, начално количество на вода **l** и множество от опасни колектори **M**, намира дали има път между двата колектора и ако има, намира онзи път между **a** и **b**, по който ще се загуби минимално количество вода и какво **количество вода** е достигнала до колектора **b**.