The basic idea:

Assume one node (for example, the onde B in the following figure) is on the max sum path. There are three kinds of paths passing node B:

(1) B and its left child path

That is, the path goes through the left side of B and B is the top end of the path. Note the path could contain only B itself if the path sum below B is less than zero.

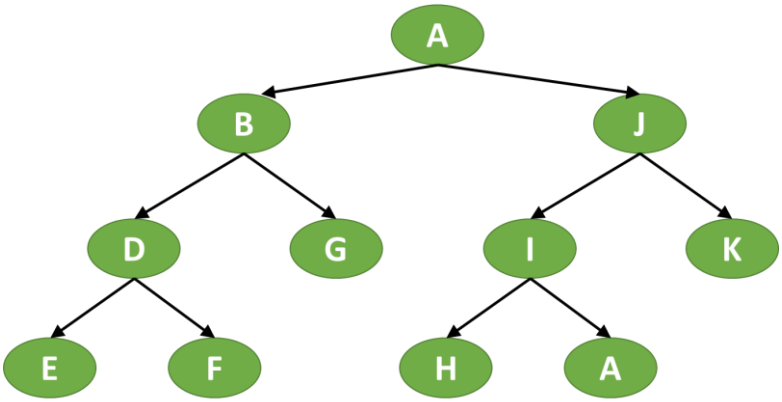
(2) B and its right child path

That is, the path goes through the right side of B and B is the top end of the path. Note the path could contain only B itself if the path sum below B is less than zero.

(3) B + right child path + left child path

That is, B is a middle node of the path.

So we could compute the 3 paths' length separately and take the largest one. Compare the largest one with the global max path length. If it is larger, then update the global max path length with it.



So the problem becomes how to compute the three paths' lengths.

To get the left side path's length, we must let its left child D tells its largest single side path length *left_child_max_len*. That is, if (D + E) > (D + F), then D tells B *left_child_max_len* = (D+ E) (of course if E and F both are less than 0, then D tells itself's value D). Now the left side path length of B is *left_path_len* = (B + *left_child_max_len*). If *left_child_max_len* < 0, then it should be B itself

(*left_path_len* = B).

To get the right side path's length, we must let its right child D tells its largest single side path length *right_child_max_len*. That is, G itself's value. Now the right side path length of B is *right_path_len* = (B + *right_child_max_len*). If *right_child_max_len* < 0, then it should be B itself (*right_path_len* = B).

To get the B+ left+ right path length, we can compute it as *combine_path_len* = *left_path_len + right_path_len – B* (because B is included in left_path_len and right_path_len twice).

Then we can take the largest one of left_path_len, right_path_len and combine_path_len to compare with the global max length.

As a recursive procedure, node B returns the max of left_path_len and right_path_len to the upper node A so that A can do the same procedure.