```c
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

typedef struct ast_node ast_node;

typedef enum op {
        ADD = 0,
        SUB = 1,
        MUL = 2,
        DIV = 3,
} op;

typedef struct ast_node {
        op operation;
        int realLeft;
        int realRight;
        ast_node *right;
        ast_node *left;
} ast_node;

void freeAst(ast_node *head){
        if (head->left != NULL)
                freeAst(head->left);
        if (head->right != NULL)
                freeAst(head->left);
        free(head);
}

int exec(ast_node *exp){
        if (exp->left != NULL){
                exp->realLeft = exec(exp->left);
                freeAst(exp->left);
                exp->left = NULL;
        }
        if (exp->right != NULL){
                exp->realRight = exec(exp->right);
                freeAst(exp->right);
                exp->right = NULL;
        }

        if (exp->operation == ADD)
                return exp->realLeft+ exp->realRight;
        if (exp->operation == SUB)
                return exp->realLeft - exp->realRight;
        if (exp->operation == MUL)
                return exp->realLeft * exp->realRight;
        if (exp->operation == DIV)
                return exp->realLeft/ exp->realRight;
        return 0;
}
```