Code:

```
#include <bits/stdc++.h>
using namespace std;

class Node
{
public:

   string data;
   Node *left = NULL, *right = NULL;
   Node(string a)
   {
      data = a;
   }
};



int x(string s)
{
   int num = 0;



     if(s[0]!='-')
       for (int i=0; i<s.length(); i++)
          num = num*10 + (int(s[i])-48);

     else
       for (int i=1; i<s.length(); i++)
       {
          num = num*10 + (int(s[i])-48);
          num = num*-1;
```

```
        }

    return num;
}



int eval(Node* root)
{

    if (!root)
        return 0;

    if (!root->left && !root->right)
        return x(root->data);

    int l_val = eval(root->left);

    int r_val = eval(root->right);



    if (root->data=="+")
        return l_val+r_val;

    if (root->data=="-")
        return l_val-r_val;

    if (root->data=="*")
        return l_val*r_val;
    if (root->data=="/")
        return l_val/r_val;
```

```
    return l_val/r_val;
}



int main()
{

    Node *root = new Node("+");
    root->left = new Node("/");
    root->left->left = new Node("*");
    root->left->left->left = new Node("2");
    root->left->left->right = new Node("3");
    root->left->right = new Node("-");
    root->left->right->left = new Node("2");
    root->left->right->right = new Node("1");
    root->right = new Node("*");
    root->right->left = new Node("5");
    root->right->right = new Node("-");
    root->right->right->left = new Node("4");
    root->right->right->right = new Node("1");
    cout <<"OUTPUT: "<< eval(root) << endl;

    return 0;
}
```

Output: