04/17/18 11:28:08 D:\git-repos\data-structure-homework\06\e17.copy.cpp

```cpp
#include <cstdio>
#include <cstring>
#include <iostream>
#define MXN 1007
using namespace std;
struct node {
    // int id;
    union { int num; char op; } dta;
    char tag;
    node *c[2];
    node *fa;
} nds[MXN];
int pri[256], totp = 0;
node *read() {
    int tot = totp++;
    char ch;
    for (ch = getchar(); ch == ' '; ch = getchar());
    if (pri[ch]) {
        nds[tot].tag = 1;
        nds[tot].dta.op = ch;
        nds[tot].c[0] = read();
        nds[tot].c[0]->fa = nds + tot;
        nds[tot].c[1] = read();
        nds[tot].c[1]->fa = nds + tot;
    } else {
        int tmp = 0;
        for (; isdigit(ch); ch = getchar()) tmp = tmp * 10 + ch - '0';
        nds[tot].tag = 0;
        nds[tot].dta.num = tmp;
        nds[tot].c[0] = nds[tot].c[1] = NULL;
    }
    return nds + tot++;
}
void dfs(node *x) {
    if (!x->tag) {
        printf("%d", x->dta.num);
        return;
    }
    if (x->fa && pri[x->fa->dta.op] > pri[x->dta.op])
        putchar('(');
    dfs(x->c[0]);
    putchar(x->dta.op);
    dfs(x->c[1]);
    if (x->fa && pri[x->fa->dta.op] > pri[x->dta.op])
        putchar(')');
}
int n;
int main() {
    pri['+'] = pri['-'] = 1;
    pri['*'] = pri['/'] = 2;
    dfs(read());
}
/**
d:\git-repos\data-structure-homework\06>g++ e17.copy.cpp

d:\git-repos\data-structure-homework\06>a
* + 1 / 2 - 3 4 + 5 6
(1+2/(3-4))*(5+6)
d:\git-repos\data-structure-homework\06>
*/
```