

UNIVERSIDADE FEDERAL DO CEARÁ

CKD109-2019.2-TO2 - ESTRUTURAS DE DADOS

AULA 05 - 2019-08-21

ALOCACÃO DINÂMICA (CONT.)

1. TESTE DO SUCESSO DA ALOCAÇÃO:

```
#include <iostream>
#include <new>
using std::cin; using std::cout; using std::nothrow;
int main ()
{
    int n; cout << "n: "; cin >> n;
    double *v = new (nothrow) double[n];
    if (v == nullptr)
    { cout << "Sem memória.\n"; return 1; }
    // USO NORMAL DE v...
    delete[] v;
}
```

AValiação de Expressões Aritméticas

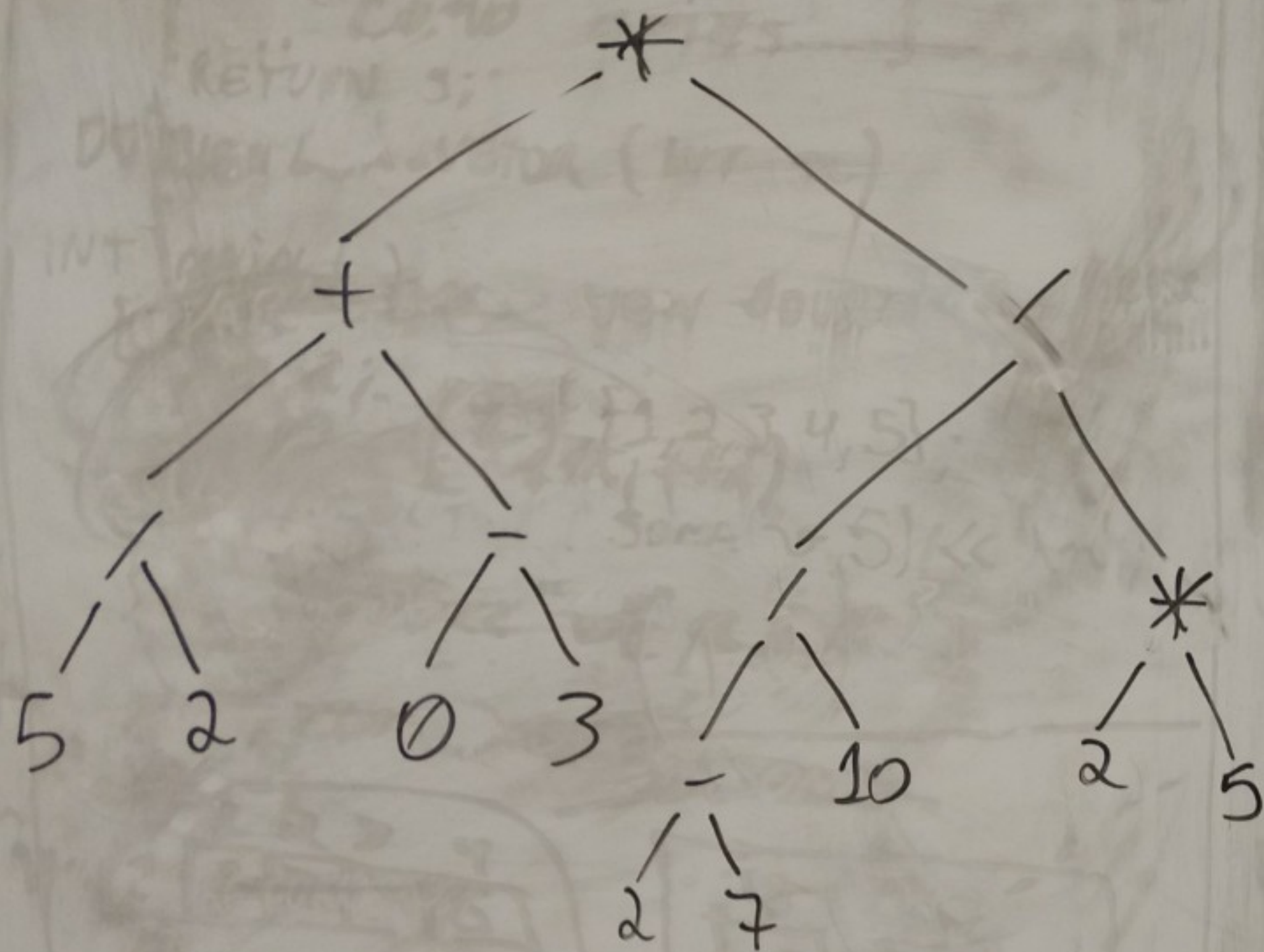
2. PROBLEMA: ESCREVA UM PROGRAMA EM C++ QUE LEIA DO USUÁRIO UMA EXPRESSÃO ARITMÉTICA ENVOLVENDO NÚMEROS NATURAIS E AS OPERAÇÕES $+$, $-$, $*$ E $/$, E QUE ESCREVA NA TELA O RESULTADO (RACIONAL) DA EXPRESSÃO. RESTRIÇÃO DA ENTRADA: A EXPRESSÃO POSSUIRÁ TODOS OS PARÊNTESES, UM PAR PARA CADA OPERADOR.

EXEMPLO DE ENTRADA:

$$\left((5/2) + (0-3) \right) * \left(((2-7)/10) / (2*5) \right)$$

O NÚMERO DE CARACTERES DA EXPRESSÃO NÃO É CONHECIDO A PRIORI. NO CASO, O PROGRAMA DEVE SOLICITAR

PAR AO USUÁRIO, NO INÍCIO DA EXECUÇÃO, A DIGITAÇÃO DE UM LIMITE SUPERIOR n PARA O NÚMERO DE CARACTERES DA EXPRESSÃO (SEM CONTAR O '\n').



ESTRATÉGIA:

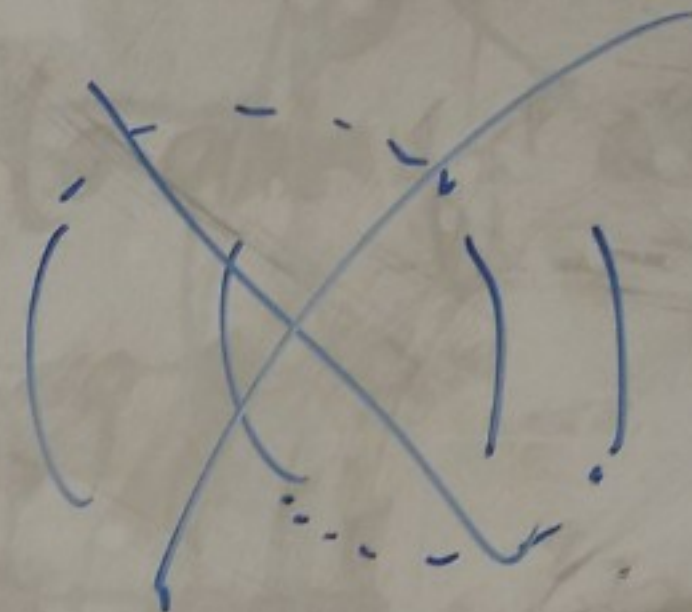
1. LER A LINHA CARACTERE-A-CARACTERE ("cin.get(c)", PARA ALGUM "char" "c").
2. PARA TRANSFORMAR O CARACTERE "c" DE UM DÍGITO "d" EM INTEIRO, BASTA FAZER $d = c - '0'$.
3. PARA TRANSFORMAR UMA SEQUÊNCIA DE DÍGITOS NUM ÚNICO INTEIRO:
$$1234 = (((1*10)+2)*10+3)*10+4$$
4. "A PRÓXIMA EXPRESSÃO A SER RESOLVIDA TERMINA NO PRÓXIMO PARÊNTESE FECHANDO".
5. FACILITA ACUMULAR NÚMEROS NUM VETOR E OPERADORES NOUTRO.

$$\cancel{5/2}$$

$$2,5 + -3$$

$$-0,5 * -0,05$$

$$\left(\frac{2,5}{(5/2)} + \left(3 * \frac{0,5}{(2/4)} \right) \right) /$$



$$-0,05$$

*

$$\left(\left(\frac{2,5}{(5/2)} + \frac{-0,5}{(-3)} \right) * \left(\frac{-0,5}{(-5/20)} / \frac{40}{(2*5)} \right) \right)$$