

题目讲解

1、求二叉树的宽度

七. (10 分) 假定某二叉树以链接形式 (每个节点包括三个字段: left, data, right) 存储, 请设计一个算法, 求该二叉树的宽度 (一棵二叉树的宽度系指: 在该二叉树的各层上, 具有结点数最多的那一层上的结点总数)。

要求:

- (1) 概要描述算法的思想;
- (2) 在关键的地方给出简明的注释;
- (3) 算法可使用 C, C++ 或 ADL 语言描述。

思路:

宽度优先搜索的变形题, 每次遍历一整层的节点数量

代码:

```
// 定义计算二叉树宽度的函数
int WidthTree(tree root){
    tree queue[maxsize];           // 使用数组模拟队列, 用于层序遍历二叉树
    int front = -1, rear = -1;     // 初始化队列头尾指针
    tree p = root;                 // 当前处理的节点
    int maxwidth = 0;              // 当前记录的最大宽度
    queue[++rear] = p;             // 根节点入队

    while(rear != front){          // 当队列不为空时, 继续遍历
        int n = rear - front;      // 当前层的节点数
        if(n > maxwidth) maxwidth = n; // 如果当前层宽度大于最大宽度, 则更新最大宽度

        // 遍历当前层的所有节点
        for(int i = 0; i < n; i++){
            p = queue[++front];     // 出队操作
            if(p->left) queue[++rear] = p->left; // 左子节点入队
            if(p->right) queue[++rear] = p->right; // 右子节点入队
        }
    }
    return maxwidth; // 返回最大层宽度
}
```

2、公式 精度求解

$$\sin(x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{(2n+1)!} \quad (-\infty < x < \infty)$$

五、【20 分】 利用公式 $\sin(x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{(2n+1)!} \quad (-\infty < x < \infty)$

编写一个求正弦函数近似值的函数，结果精确到 10^{-5} 。实现过程中，不允许调用任何库函数，但允许调用自己定义并实现的其它函数。

思路：

如何精确到结果？

当计算每一项的时候，当出现一项结果小于这个精度的时候，就可以跳出循环，返回结果。

代码：

```
#include <stdio.h>

// 函数功能：计算阶乘 n!
double factorial(int n) {
    double result = 1.0;
    for (int i = 2; i <= n; ++i) {
        result *= i;
    }
    return result;
}

// 函数功能：计算x的n次幂
double power(double x, int n) {
    double result = 1.0;
    for (int i = 0; i < n; ++i) {
        result *= x;
    }
    return result;
}

// 函数功能：计算绝对值
double absolute(double x) {
    return x > 0 ? x : -x;
}

// 函数功能：根据泰勒级数求正弦函数近似值
double sin_approximation(double x, double precision) {
    double term, sum = 0.0;
    int sign = 1;

    // 循环累加每一项直到精度要求满足
    for (int n = 0; ; n++) {
        term = sign * power(x, 2 * n + 1) / factorial(2 * n + 1);
        sum += term;
        if (absolute(term) < precision) break; // 如果当前项小于指定精度，则停止迭代
        sign = -sign; // 更新符号
    }
}
```

```
}

    return sum;
}

int main() {
    double x;
    printf("请输入角度（弧度制）：");
    scanf("%lf", &x);

    // 计算sin(x)的近似值，精确到10^-5
    double sin_value = sin_approximation(x, 1e-5);
    printf("sin(%f) ≈ %f\n", x, sin_value);

    return 0;
}
```