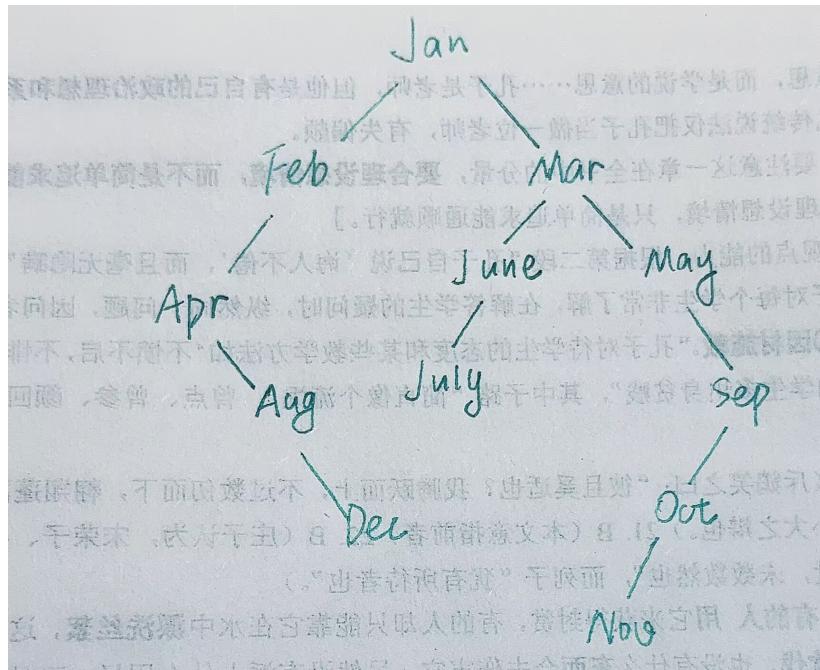


7.2.3

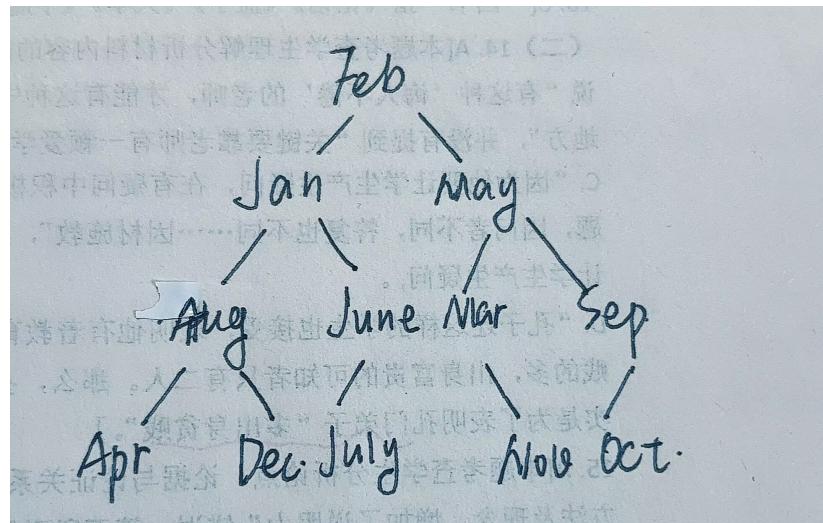
1. $42/12=3.5$



2. Apr, Aug, Dec, Feb, Jan, June, July, Mar, May, Nov, Oct, Sep

37/12=3.1

3. 37/12=3.1



7.3.5

```

typedef struct AVLNode {
    int key;
    int b; // 平衡因子 = 右子树高度 - 左子树高度
    struct AVLNode *lchild, *rchild;
} AVLNode, *AVLTree;
  
```

// 算法：利用平衡因子求平衡二叉树的高度

```

int getAVLHeight(AVLTree T) {
    if (T == NULL) return 0; // 空树高度为0
  
```

```
int height = 1; // 当前路径高度
AVLNode *p = T;

while (p != NULL) {
    if (p->b >= 0) {
        // 右子树高度 >= 左子树高度
        if (p->rchild != NULL) {
            height++;
            p = p->rchild;
        } else {
            break; // 到达叶子节点
        }
    } else {
        // 左子树高度 > 右子树高度
        if (p->lchild != NULL) {
            height++;
            p = p->lchild;
        } else {
            break; // 到达叶子节点
        }
    }
}

return height;
}

// 辅助函数：创建新结点
AVLNode* createNode(int key, int b) {
    AVLNode node = (AVLNode)malloc(sizeof(AVLNode));
    node->key = key;
    node->b = b;
    node->lchild = NULL;
    node->rchild = NULL;
    return node;
}
```