2022北邮809数据结构答案（非官方，仅供参考）

一、填空题

1. 时间复杂度　空间复杂度
2. $O(log_2(n))$
3. $(n-1)/2$
4. 4
5. $i*n*p+p*j+k$　$m*n*k+m*j+i$
6. $(rear+1)\%n==front$　$front==rear$　$(rear-front+n)\%n$
7. 92　1
8. 39
9. 100　1　99　8
10. $FEGKJIHDCBA$
11. $n+1$
12. e　$2e$
13. 10
14. 10.5　10.95
15. 1
16. 右子树　左子树
17. 7　3.15
18. 中序
19. 简单选择排序　0

二、单选题

1. D
2. B
3. D
4. B
5. C
6. C
7. C
8. C
9. C
10. A
11. C
12. B
13. D
14. A
15. C
16. D
17. A
18. C

19. D
20. B
21. D
22. B
23. B
24. C
25. D
26. B

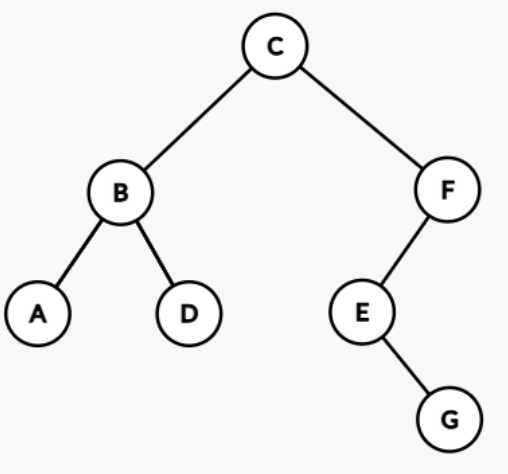
三、简答题

1.

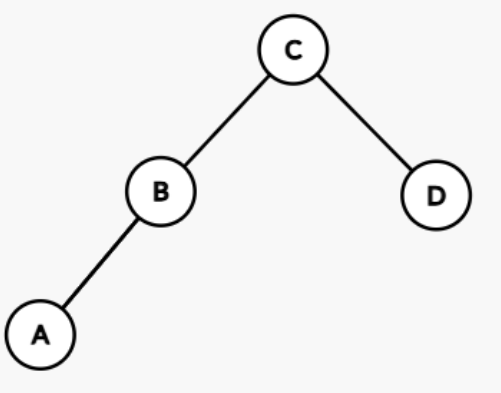
(1)寻找单链表的最后一个结点
(2)将单链表的第一个结点作为新链表的最后一个结点
(3)$(a_2, \ldots, a_n, a_1)$

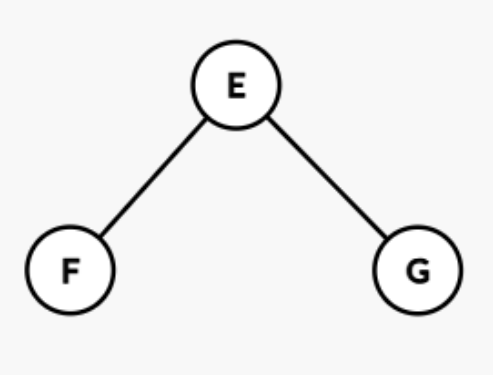
2.

(1)先序序列：$CBADFEG$
中序序列：$ABDCEGF$
后序序列：$ADBGEFC$
(2)



(3)



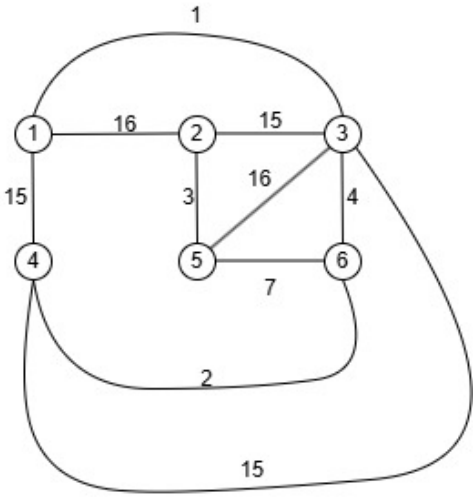


3.

```
if(R && i<k)
InOrder(R->lch, k)
if(i == k)
InOrder(R->rch, k)
```

4.

(1)

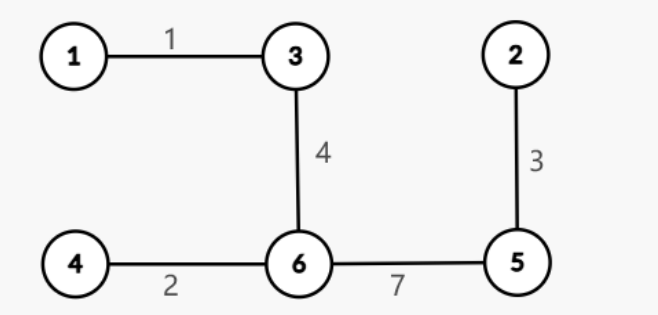


(2)

深度优先搜索：$1,2,3,4,6,5$

广度优先搜索：$1,2,4,3,5,6$

(3)



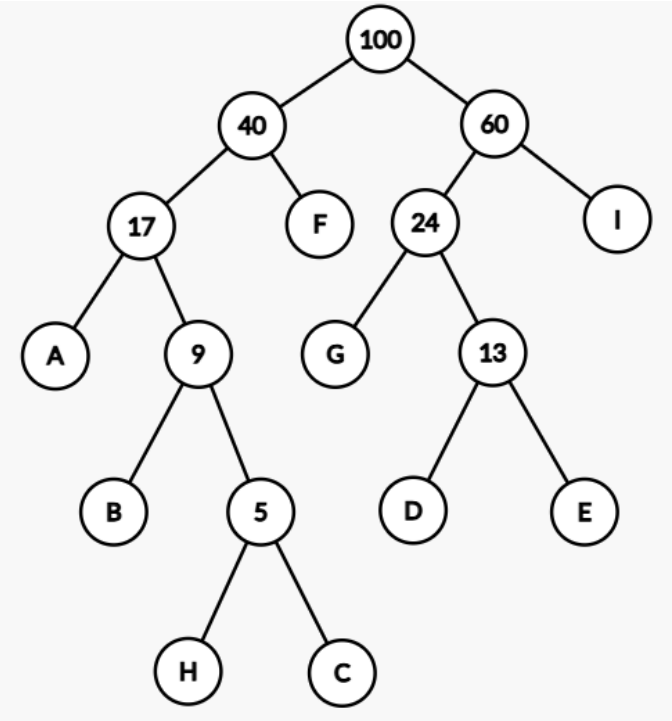
(4)

$v2:1,3,6,5,2 \quad 15$

$v3: 1, 3 \quad 1$
$v4: 1, 3, 6, 4 \quad 7$
$v5: 1, 3, 6, 5 \quad 12$
$v6: 1, 3, 6 \quad 5$

5.

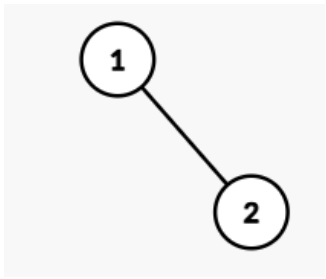
(1) 哈夫曼编码
(2)

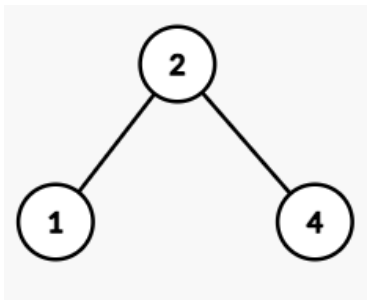


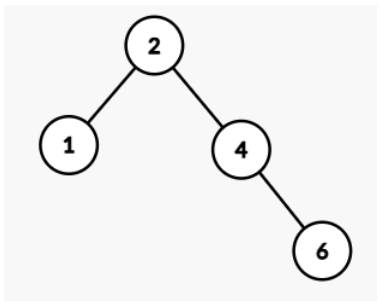
$A: 000$
$B: 0010$
$C: 00111$
$D: 1010$
$E: 1011$
$F: 01$
$G: 100$
$H: 00110$
$I: 11$

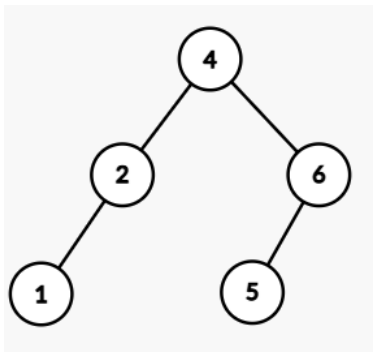
$8 * 3 + 4 * 4 + 3 * 5 + 6 * 4 + 7 * 4 + 23 * 2 + 11 * 3 + 2 * 5 + 36 * 2 = 268$

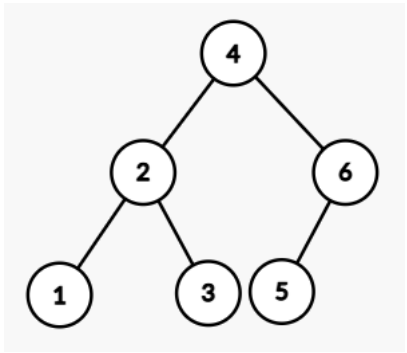
6.











7.

(1) 18

(2)

	Apr	Aug	Sep	Dec		Feb				Jan	Jun	Jul	Mar	May	Oct	Nov	

(3)
$(1+1+1+1+2+2+3+2+1+1+3+1)/12=1.58$

7.

(1)$70,70,15,18,60,11,7,12,13,25$
(2)$70,60,15,18,25,11,7,12,13,70$
(3)$7,13,11,12,18,15,60,25,70,70$
(4)$12,13,11,70,15,25,7,18,60,70$
(5)$7,13,11,70,25,15,12,18,70,60$
(6)$7,11,12,70,25,15,13,18,70,60$
(7)$7,12,13,11,70,25,15,18,60,70$

四、程序题

1.

`p->data`
`p->next`
`p->data`
`q->next`
`ai`

2.

```
for(int i=0; i<n; i++){
        B[A[i]]++;
}
```

```
for(int i=1; i<=K; i++){
        B[i] += B[i-1];
}
```

```
for(int i=n-1; i>=0; i--){
        C[B[A[i]]-1] = A[i];
        B[A[i]]--;
}
```

3.

```
void TopK(int a[], int b[], int n, int K) {
    for (int i = 0; i < K; i++) {
        b[i] = a[i];
    }

    for (int i = 1; i < K; i++) {
```

```
        int key = b[i];
        int j = i - 1;
        while (j >= 0 && b[j] > key) {
            b[j + 1] = b[j];
            j--;
        }
        b[j + 1] = key;
    }

    for (int i = K; i < n; i++) {
        if (a[i] > b[0]) {
            b[0] = a[i];
            int key = b[0];
            int j = 1;
            while (j < K && b[j] < key) {
                b[j - 1] = b[j];
                j++;
            }
            b[j - 1] = key;
        }
    }
}
```

4.

```
struct BiNode {
    int data;
    BiNode* lch;
    BiNode* rch;
    BiNode* parent;
};

void Path(BiNode* root, BiNode* PA, BiNode* PB) {
    vector<BiNode*> pathA, pathB;
    BiNode* p = PA;
    while(PA){
        pathA.push_back(PA);
        PA = PA->parent;
    }

    while(PB){
        pathB.push_back(PB);
        PB = PB->parent;
    }

    int i = pathA.size() - 1, j = pathB.size() - 1;
    while (i >= 0 && j >= 0 && pathA[i] == pathB[j]) {
        i--;
        j--;
    }

    cout << "Path from PA to PB: ";
```

```
    for(int k=i+1; k<pathA.size(); k++){
        cout << pathA[k]->value << " ";
    }
    for(int k=j; k>=0; k--){
        cout << pathB[k]->value << " ";
    }
}
```