

安徽大学 20 22 —20 23 学年第 2 学期

《数据结构》考试试卷 (A 卷) 参考答案及评分标准
(闭卷 时间 120 分钟)

一、算法阅读题 (每小题 5 分, 共 20 分)

1. $O(n^3)$ (5 分)

2. $O(1)$ (5 分)

3. $q \rightarrow next = p \rightarrow next;$ (3 分) $p \rightarrow next = q;$ (2 分)

4.

$like(t1 \rightarrow lchild, t2 \rightarrow lchild)$ (3 分)

$like(t1 \rightarrow rchild, t2 \rightarrow rchild)$ (2 分)

二、简答题 (每小题 5 分, 共 20 分)

5.

$7 * 11 * 4 = 308$ (3 分)

1012 (2 分)

6. 求下列广义表的表长、表头和表尾, 结果填于表 1 中。(每空 1 分)

(1) $A = ((a, b), c, ((d)))$

(2) $B = ((a, b, c), x, (u, t, w))$

表 1

广义表	表长	表头	表尾
$A = ((a, b), c, ((d)))$	3	(a, b)	$(c, ((d)))$
$B = ((a, b, c), x, (u, t, w))$	3	(a, b, c)	$(x, (u, t, w))$

7.

栈满的条件: $top[1] + 1 == top[2]$ (3 分)

栈 1 空的条件: $top[1] == -1$ (1 分)

栈 2 空的条件: $top[2] == m + 1$ (1 分)

8. BCDEF EF (5 分)

三、应用题 (每题 10 分, 共 30 分)

9.

答案:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
57	32	59	41	98	47	94	50	63	52	42	54	60	72	86	90	(3 分)
57	32	42	41	60	47	86	50	63	52	59	54	98	72	94	90	(2 分)
57	32	42	41	60	47	59	50	63	52	86	54	98	72	94	90	(3 分)
42	32	57	41	59	47	60	50	63	52	86	54	94	72	98	90	(2 分)
32	41	42	47	50	52	54	56	59	60	63	72	86	90	94	98	

10.

(1) (5分)

0	1	2	3	4	5	6	7	8	9	10	11	12
55	12	23	80	46	38	27		8	20	10		
1	1	2	1	3	1	2		1	1	1		

(2) $ASL=14/10=1.4$ (5分)

11.

(1) 列出事件的最早发生时间。(6分)

事件	1	2	3	4	5	6	7
最早发生时间	0	5	16	9	25	40	54

(2) 指出关键路径。(4分)

1-3-5-6-7

四、算法设计题 (每小题 10 分, 共 30 分)

12.

答案:

```

int Binsch(ElemType A[], int low, int high, KeyType K)
{
    if(low<=high) (1 分)
    {
        int mid=(low+high) / 2; (2 分)
        if(K==A[mid].key) return mid; (2 分)
        else if (K<A[mid].key) return Binsch(A,low,mid-1,K); (2 分)
        else return Binsch(A,mid+1,high,K); (2 分)
    }
    else return -1; (1 分)
} Binsch

```

13.

(1)

struct node

```

{   datatype data;
    struct node * prior,* next;
};

```

(2 分)

```

(2) void change(Dullinklist &L, node *p)
{
    node *q;
    q->prior->next=p;
    p->next->prior=q;
    p->prior=q->prior;
    q->next=p->next;
    p->next=q;
    q->prior=p;
}

```

(4 分)

```

(3)
p->prior->next= p->next;
p->next->prior = p->prior;

```

(4分)

14.

```

(1) int depth(node *t)    (5 分)
{
    int num1,num2;
    if(t==NULL) return(0);
    if(t->lchild ==NULL&& t->rchild ==NULL) return(1);
    else
    {
        num1=depth(t->lchild );
        num2=depth(t->rchild );
        if(num1>num2)
            return(num1+1);
        else
            return(num2+1);
    }
}

```

```

(2) void lev_traverse(node *T)    (5 分)
{
    node *q[100],*p;
    int head,tail;
    q[0]=T;head=0;tail=1;
    while(head<tail)
    {
        p=q[head++];
        printf("%c",p->data);
        if(p->rchild!=NULL)
            q[tail++]=p->rchild;
        if(p->lchild!=NULL)
            q[tail++]=p->lchild;
    }
}

```