

杭州电子科技大学

2018 年攻读硕士学位研究生招生考试

《数据结构与组成原理》试题

(试题共八大题, 共 8 页, 总分 150 分)

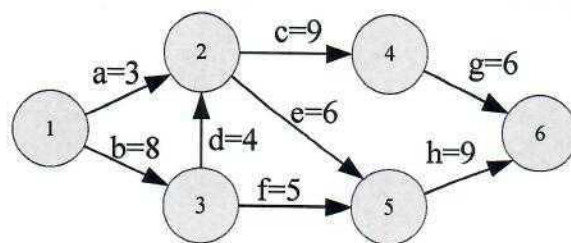
姓名_____报考专业_____准考证号_____

【所有答案必须写在答题纸上, 做在试卷或草稿纸上无效!】

一、单项选择题 (本大题共 7 小题, 每小题 2 分, 本大题共 14 分)

1. 若将长度为 n 的循环单链表链接在长度为 m 的循环单链表之后, 其算法时间复杂度为 ()。
A: $O(1)$ B: $O(n)$ C: $O(m)$ D: $O(m+n)$
2. 用单链表方式存储队列 (有头尾指针, 非循环), 在进行删除运算时 ()。
A: 仅修改头指针 B: 仅修改尾指针
C: 头、尾指针都须修改 D: 头、尾指针可能都要修改
3. 以下说法中 () 是正确的。
A: 构造线索二叉树是为方便找到每个结点的双亲。
B: 任何一棵二叉树, 终端结点数为度为 2 的结点数减 1。
C: 二叉树不能用顺序结构表示。
D: 完全二叉树中, 某叶子结点的双亲若存在左兄弟 X , 则 X 一定不是叶子结点。
4. 若邻接表中有奇数个表结点, 则该图是 ()。
A: 连通图 B: 强连通图
C: 无向图 D: 有向图
5. 以下叙述中, 不正确的是 ()。
A: 图和树的区别之一在于树的序偶对个数等于顶点数减一, 而图的序偶对个数可大于顶点数。
B: 假设有图 $G=\{V, E\}$ 及 $G'=\{V', E'\}$, 满足 $V'\subseteq V$ 且 $E'\subseteq E$, 则 G' 是 G 的子图。
C: 无向图的连通分量指无向图中的极小连通子图。
D: 连通图的遍历一定能从图中某一顶点出发访遍图中全部顶点。

6. 下列 AOE 网表示一项包含 8 个活动的工程。其关键活动为 ()。



- A: a,b,c,d,e,f B: b,c,d,e,h,g
C: b,c,d,e,f,h D: a,b,d,e,f,h

7. 下列排序方法中, 时间性能与待排序记录的初始排列无关的是 ()。

- A: 插入排序和快速排序 B: 归并排序和快速排序
C: 堆排序和归并排序 D: 插入排序和归并排序

二、填空题 (本大题共 5 小题, 共 7 个空, 每空 2 分, 本大题共 14 分)

- 若每一数据只能进栈一次、出栈一次, 则通过栈对先后输入的三个数据 3,2,1 进行调度, 不能得到的调度结果是: _____。
- 设有二维数组 $A[10][10]$, 假设每个数组元素用 4 个字节存储, 存储器按字节编址, 从首地址 1000 开始以行序为主次序连续存放, 则数组元素 $A[3][7]$ 的起始地址为 _____。
- 广义表 $((), (a), ((b, c), d))$ 的表尾为: _____。
- 假设用于通讯的电文仅由 8 个字母组成, 字母在电文中出现的频率分别为 7, 19, 2, 6, 32, 3, 21, 10。若为这 8 个字母设计哈夫曼编码(设合并两颗子树时次序在前的为左子树, 次序在后的为右子树)。则频率为 6 的字母编码为: _____, 频率为 7 的字母编码为: _____。
- 哈希表的地址区间为 0-10, 哈希函数取为 $H(K)=K \bmod 11$ 。采用开放地址法处理冲突, 增量序列取平方探测再散列, 将关键字序列 26,25,72,38,8,18, 59,32,28 依次存储到哈希表中。则关键字 59 存放在表中的下标地址是____, 关键字 28 存放在表中的下标地址是_____。

三、问答与图示题 (本大题共 4 题, 每题 10 分, 本大题共 40 分)

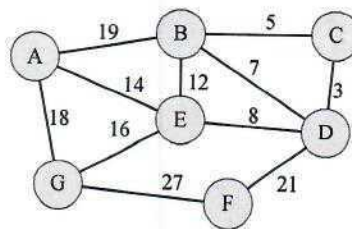
1. 已知森林的先序次序为: A,B,C,D,E,F,G,H,I,J,K。

中序次序为: B,E,F,C,D,A,G,I,K,J,H。

- 画出该森林。(本小题 4 分)
- 利用孩子-兄弟法将其转化二叉树。(本小题 3 分)
- 将该二叉树的中序线索化。(本小题 3 分)

2. 已知某无向网如图所示:

- (1) 采用邻接多重表表示该无向网, 画出存储结构示意图 (要求符合按边结点次序插入在相应链表中)。(本小题 4 分)



- (2) 依据所建图的邻接多重表存储, 分别给出该图的深度优先和广度优先遍历序列。(本小题 2 分)
- (3) 分别按 Prim 算法 (从顶点 A 始) 及 Kruskal 算法依次给出最小生成树的边集。(本小题 4 分)

3. 依次在查找表中插入: 76, 29, 33, 71, 26, 19, 45, 23, 67, 18, 51

- (1) 依照插入次序建立二叉排序树。(本小题 2 分)
- (2) 依照插入次序建立平衡的二叉排序树。(本小题 4 分)
- (3) 依照插入次序建立 3 阶 B 树。(本小题 4 分)

4. 给出一组关键字 $T=(37, 58, 31, 45, 8, 25, 32, 19, 26, 18, 62, 11)$:

- (1) 给出希尔排序 (第一趟排序的增量为 5) 一趟排序的结果。(本小题 2 分)
- (2) 给出快速排序 (选第一个记录为枢轴) 一趟排序的结果。(本小题 2 分)
- (3) 给出小顶堆的排列结果。(本小题 2 分)
- (4) 给出归并排序 (非递归) 一趟排序的结果。(本小题 2 分)
- (5) 给出基数排序一趟排序的结果。(本小题 2 分)

四、程序阅读题 (阅读以下函数, 指出算法的功能。本大题共 3 小题, 每小题 5 分, 本大题共 15 分)

1.

```
bool A1(HString &S, int pos, HString T) {
    if (pos < 1 || pos > S.length + 1)    return false;
    if (T.length) {
        S.ch = (char*)realloc(S.ch, (S.length + T.length) * sizeof(char));
        if (!S.ch)
            return false;
        for (i = S.length - 1; i >= pos - 1; i--)
            S.ch[i + T.length] = S.ch[i];
        for (i = 0; i < T.length; i++)
            S.ch[pos - 1 + i] = T.ch[i];
        S.length += T.length;
    }
    return true;
}
```

2.

```

bool A2 (Bitree T) {
    Initstack (S) ;
    Push (S, T);
    while (!StackEmpty(S) ) {
        while (GetTop (S, p)&& p ) {
            printf("%c",p->data);
            push (S, p->lchild )
        }
        Pop (S,p);
        if (! StackEmpty(S) ) {
            Pop(S, p) ;
            Push (S, p->rchild );
        }
    }
    return true ;
}

```

3.

```

bool A3-1( Graph G, VertexType Vi, VertexType Vj ) {
    for (v=0; v<G.vexnum; ++v)
        visited[v] = false;
    found = false;
    i = LocateVex(G, Vi);
    j = LocateVex(G, Vj);
    A3-2(G, i, j, found);
    if (found)
        return true;
    else
        return false;
}

void A3-2(Graph G, int v, int u ,bool &found) {
    visited[v] = true;
    for(w=FirstAdjVex(G, v); ! found && w>=0; w=NextAdjVex(G, v, w)){
        if( w == u){
            found = true;
            break;
        }
        if (!visited[w]) A3-2(G, w, u, found);
    }
}

```

五、算法设计题（本大题共 2 题，本大题共 22 分）

1. （本题 10 分）已知无表头结点的单链表 la 及单链表 lb 存在，写一算法，删除单链表 la 中第 i 个结点起长度为 len 的结点，并将其插入至单链表 lb 第 j 个结点之前。
2. （本题 12 分）已知有向图采用邻接表存储表示，试用深度优先搜索的策略基于图的邻接表存储写一算法，判断有向图是否存在回路。

六、计算题（本大题共 3 题，本大题共 9 分）

某机浮点数为 16 位，阶码 8 位，尾数 8 位，各含一位符号位，均以补码表示，格式如下：

数符（1 位）	阶码（8 位）	尾数数值（7 位）
---------	---------	-----------

1. （本题 2 分）将实数 X 按照上述格式表示为规格化浮点数，以十六进制表示为 81BCH，请写出 X 的十进制真值。
2. （本题 2 分）另一实数 $Y=39/128$ ，请按照上述格式将其表示为规格化浮点数，并以十六进制表示。
3. （本题 5 分）计算上述 X 和 Y 的和 $[X+Y]_{浮}$ ，模拟计算机浮点运算器，列出计算过程（用补码计算，采用 0 舍 1 入法舍入）。

七、分析设计题（本大题共 3 题，本大题共 16 分）

某 16 位计算机的 CPU 地址总线 20 位，数据总线 16 位，存储器按字编址。CPU 有 M/\overline{IO} 信号，=1 访问存储器，=0 访问外设； R/\overline{W} 信号，=1 执行读操作，=0 执行写操作：

1. （本题 2 分）该 CPU 能支持的最大物理主存储器的容量是多少？
2. （本题 4 分）在某个应用场景中，需要为该 CPU 配置一个 32KB 的 ROM，图 1 是其设计图，其中 74LS138 是 3:8 译码器芯片，功能表如表 1。请写出该 ROM 的地址范围。

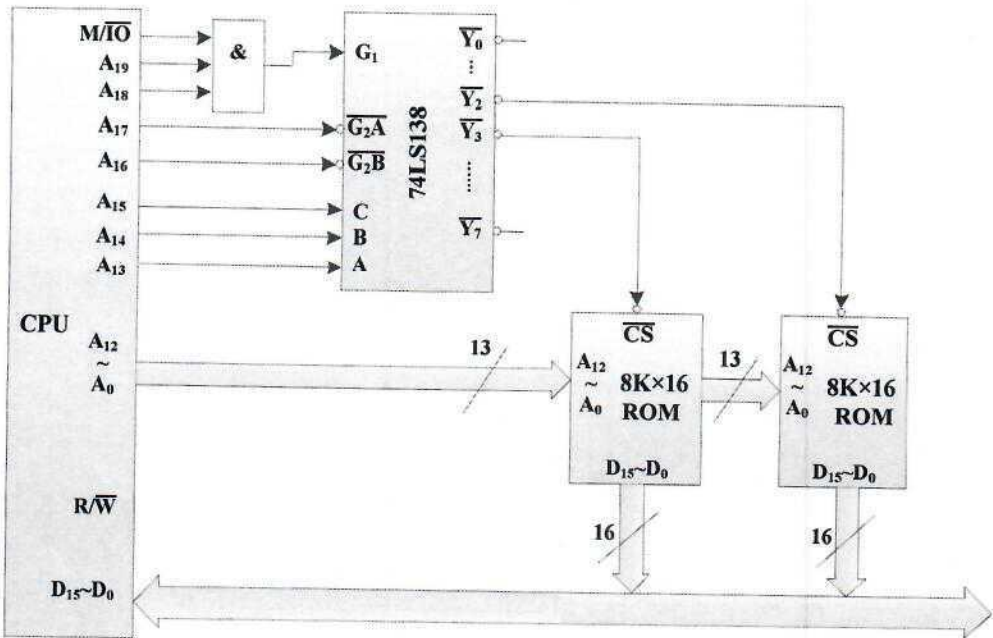


图 1 CPU 与 32KB ROM 的连接图

表 1 74LS138 译码器功能表

输入端						输出端							
G_1	$\overline{G_2A}$	$\overline{G_2B}$	C	B	A	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$	$\overline{Y_4}$	$\overline{Y_5}$	$\overline{Y_6}$	$\overline{Y_7}$
0	x	x	x	x	x	1	1	1	1	1	1	1	1
x	1	x	x	x	x	1	1	1	1	1	1	1	1
x	x	1	x	x	x	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	0	0	1	1	0	1	1	1	1	1	1
1	0	0	0	1	0	1	1	0	1	1	1	1	1
1	0	0	0	1	1	1	1	0	1	1	1	1	1
1	0	0	1	0	0	1	1	1	1	0	1	1	1
1	0	0	1	0	1	1	1	1	1	1	0	1	1
1	0	0	1	1	0	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	1	1	1	1	0

3. （本题 10 分）在另一个应用场景中，需要为该 CPU 配置一个容量为 64K×16 位的 SRAM，起始地址为 00000H，可用的芯片有：16K×8 位的 SRAM，芯片有片选信号 \overline{CS} 和写使能信号 \overline{WE} ，均为低电平有效；74LS138 芯片。请使用这两种芯片以及基本逻辑门扩展成所需存储器，并画出与 CPU 的连接图。

八、综合设计题（本大题共 4 题，本大题共 20 分）

某 16 位模型计算机的系统结构如图 2 所示，CPU 中含 8 个通用寄存器；所有的控制信号均为高电平有效，控制信号的下标 in 表明将数据打入部件，out 表明将部件数据输出。譬如 LC_{in} 表明将运算器的结果 F 送入 LC， LC_{out} 表明将 LC 的内容送到数据总线 DB 上。

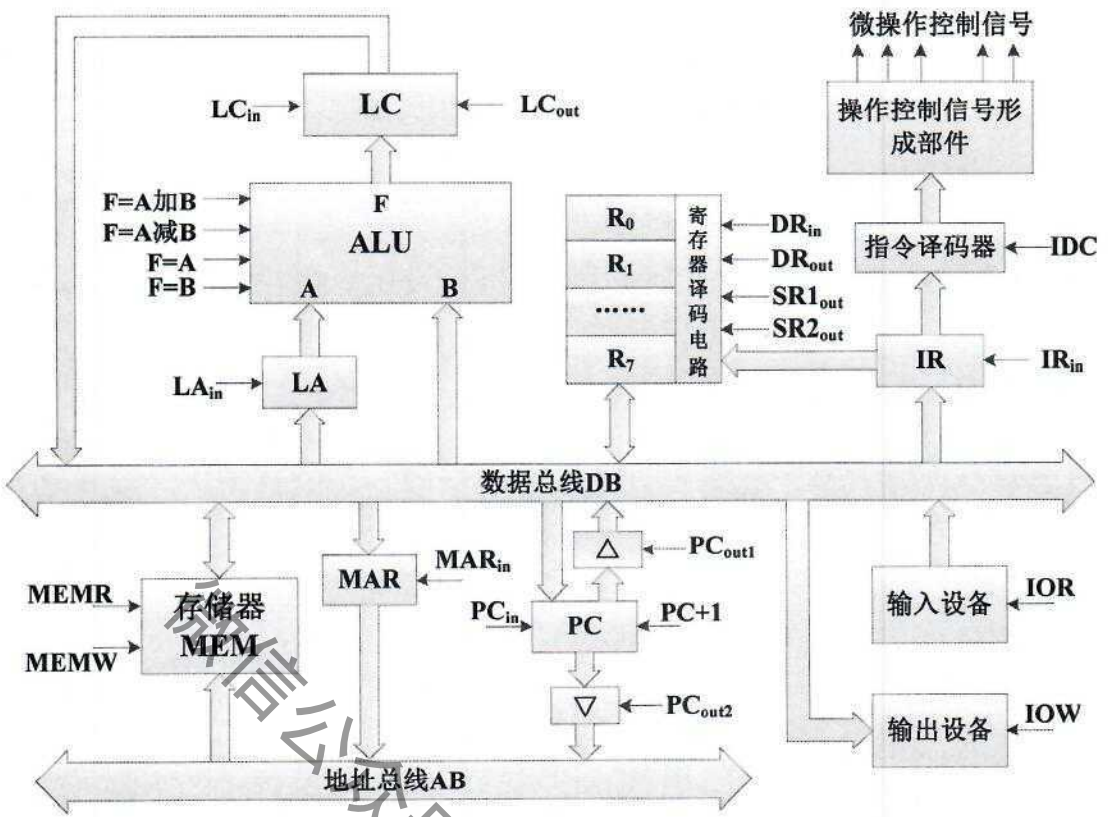


图 2 模型计算机的系统结构图

- （本题 8 分）假设指令系统有 8 种寻址方式，在指令中以单独的寻址方式码字段指明；如果指令系统中需要设置三地址指令（R-R-R 型）、两地址指令（R-R 型）和单地址指令（R 型），请设计一个单字指令可行的方案，并分别计算指令系统中，三地址指令、两地址指令和单地址指令最多有多少条？
- （本题 5 分）某条指令的执行过程如图 3 所示，其中， $MEM[PC]$ 表示以 PC 为地址读出存储器 MEM 的内容。请分析该指令的具体功能、源操作数和目的操作数的寻址方式，以及指令含有几个字。

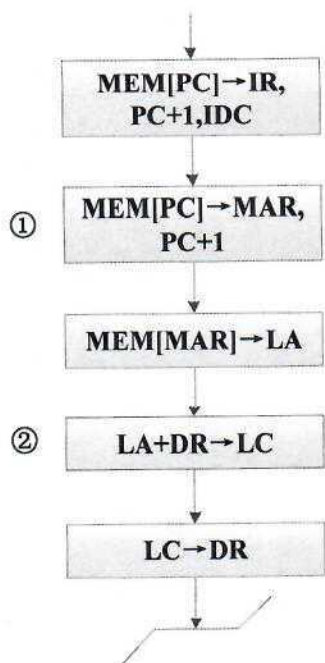


图3 指令流程图

3. （本题 3 分）假设图 3 中的每一个方框为一条微指令，则执行微指令的过程就是发送一组微操作控制信号序列。譬如，参照图 2，图 3 中微指令①所要发送的微操作控制信号序列是： PC_{out2} ，MEMR， MAR_{in} ，PC+1。请分析要实现微指令②，必须要发送的微操作控制信号序列。
4. （本题 4 分）假设该 CPU 的控制器采用微程序控制器，下址字段 9 位；后继微地址的转移方式有 3 种（含 IDC 信号），采用编译码方式；控制字段采用直接控制法。试设计其微指令格式，计算控制存储器的容量。

