

中国科学院大学

2020 年招收攻读硕士学位研究生入学统一考试试题

科目名称：计算机专业综合

考生须知：

1. 本试卷满分为 150 分，全部考试时间总计 180 分钟。
2. 所有答案必须写在答题纸上，写在试题纸上或草稿纸上一律无效。
3. 试卷共 16 道大题，每题 15 分，共 240 分，考生可以任意选择其中 10 道大题回答，并在答题纸的该题答案前标明“选做本题”。
4. 如果选做的题目多于 10 道，则判卷将按照所选做试题的题号顺序选择前 10 道大题计分，后续所做视为无效考试内容。

第一部分：《数据结构》

第一题，简答题，共 15 分

- (1) 请给出下列程序的时间复杂度 ($n > 0$)。(2 分)

```
void func(int n) {  
    int i,j;  
    for(i=1,j=0; j<=n; j=j+i) i++;  
}
```

- (2) 现有一个线性表的应用，线性表的元素总数不确定，其主要的操作是插入元素、删除表尾元素、查找表尾结点和查找指定结点的前驱结点，那么请问用怎样的数据结构比较好？请给出理由和该数据结构的 C 语言定义。(3 分)

- (3) 若循环队列存储在数组 $A[0..m]$ ，头指针 $Front$ 指向当前队头元素，尾指针 $Rear$ 指向当前队尾元素的下一个位置，那么当前实际存储有多少队列元素？现有元素 x 需要入队，请写出元素入队的语句。(3 分)

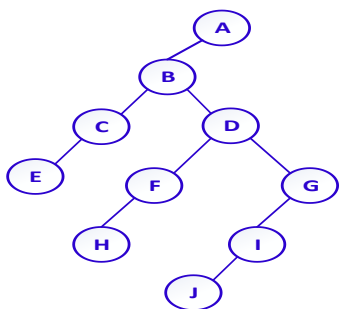
(4) 用如下数据结构存储广义表:

```
typedef enum {ATOM, LIST} ElemTag;
typedef struct GLNode {
    ElemTag tag;
    union {
        AtomType atom;
        struct {struct GLNode *hp, *tp;} ptr;
    }
} *Glist;
```

那么, 对于广义表 $X=(A,(),(B,C)),(D,E))$, 给出其存储结构图, 并利用 Head、Tail 操作分离出元素 E。(3 分)

(5) 现在需要从 5000 个元素组成的序列中, 用最快的速度挑出前 10 个最大的元素。请在插入排序、快速排序、堆排序、归并排序、基数排序中选择一个最合适的排序方法, 并说明理由。(4 分)

第二题, 问答题, 共 15 分。对于下图的树:



- (1) 求该树的先序遍历、中序遍历的结果, 画出该树的后序线索树;(7 分)
- (2) 假设二叉树按以下定义的二叉链表进行存储, 请设计程序, 判断一颗二叉树是不是完全二叉树。(8 分)

```
typedef struct BiTNode {
    TElemType data;
    struct BiTNode *lchild, *rchild;
} BiTree;
```

第三题, 问答题, 共 15 分

- (1) 已知有向图 $G=(V,E)$, 其中 $V=\{v_1, v_2, v_3, v_4, v_5, v_6\}$, $E=\{\langle v_1, v_2 \rangle, \langle v_2, v_3 \rangle, \langle v_3, v_4 \rangle, \langle v_5, v_2 \rangle, \langle v_5, v_6 \rangle, \langle v_6, v_4 \rangle\}$, 请列出图中全部可能的拓扑排序序列;(5 分)
- (2) 已知有向图, 在其邻接矩阵表示中, 主对角线以下的元素均为 0, 那么该图的拓扑排序序列是否存在? 如存在, 是否唯一? (2 分)

(3) 假设有向图 g 采用如下的存储结构, 设计算法, 用于在 g 上找到从一个顶点(设为 S)到另一个顶点(设为 D)的一条简单路径, 并将该路径记录在 $\text{char} *Path$ 中。(8分)

```
typedef char ElemType;
typedef struct node {
    int vindex;    //邻接点在表头结点数组中的位置(下标)
    struct node *next; //指向下一个表结点
} NodeLink; // 表结点类型定义

typedef struct {
    int vexnum, edgenum, kind; //分别记录图的顶点数、边数、种类标志
    struct {
        ElemType vertex;
        NodeLink *first; // 指向第一个表结点
    } v[MAX_VERTEX_NUM];
} AGraph;
```

第四题, 问答题, 共 15 分

(1) 面对 n 个数据元素, 进行顺序查找、折半查找、哈希查找的时间复杂度分别是 $O(n)$ 、 $O(\log n)$ 、 $O(1)$, 这是否说明低效的方法就不再需要? (3分)

(2) 如何衡量哈希函数的优劣? (4分)

(3) 设哈希表长度为 15, 哈希函数为 $H(\text{key}) = \text{key} \bmod 13$, 解决冲突的方法为线性探测法, 针对关键字序列 {6, 18, 21, 24, 45, 20, 55, 27, 70, 11, 10}, 给出对应的哈希表存储, 若查找关键字 45, 需要依次与哪些关键字进行比较? 求出等概率下查找成功的平均查找长度。(8分)

第二部分: 《操作系统》

第五题, 问答题, 共 15 分

在单 CPU 和两台 I/O 设备 (I1、I2) 的多道程序设计环境下, 同时投入 3 个作业 Job1、Job2、Job3 运行。这 3 个作业对 CPU 和输入/输出设备的使用顺序和时间如下:

Job1: I1 (20ms); CPU (10ms); I1 (20ms); CPU (10ms); I1 (20ms)

Job2: I2 (30ms); CPU (20ms); I1 (20ms)

Job3: CPU (40ms); I2 (10ms); CPU (10ms)

假设 CPU 和 I/O 设备之间、两台 I/O 设备之间都能并行工作, Job1 优先级最高, Job2 次之, Job3 优先级最低, 优先级高的作业可以抢占优先级低的作业的 CPU, 但不抢占 I1 和 I2。试求:

- (1) 3 个作业从投入到完成分别需要的总时间和等待时间；
- (2) CPU 的利用率；
- (3) 两个 I/O 设备的利用率。

第六题，问答题，共 15 分

某多处理器系统中，两个线程 A、B 通过一段连续的共享内存区 4K 字节进行通信，其中 A 为发送者，B 为接收者，每次通信的消息固定为 64 字节。试设计数据结构和算法，保证通信的正确性和效率。请给出数据结构及 send/recv 的伪代码（要求控制数据也存放在共享缓冲区）。

第七题，问答题，共 15 分

一个采用硬件页表的 32 位操作系统的虚拟地址由四部分组成：

4-bit | 8-bit | 8-bit | 12-bit

该系统使用三级页表，最高的 4 位为第一级，以此类推。每个页表项需 4 字节。试回答如下问题：

- (1) 该系统的页大小是多少？
- (2) 一个进程占用从地址 0x0 开始的 512 KB 内存，求分配给该进程的页表大小。
- (3) 一个进程的代码段起始地址为 0x00000000，大小为 72 KB；数据段起始地址为 0x40000000，大小为 3 MB；栈段起始地址为 0xE0000000，从低地址向高地址生长，当前大小为 128 KB。求分配给该进程的页表大小。

第八题，问答题，共 15 分

某个类 FFS 文件系统中每个 inode 有 10 个直接指针，4 个一级间接指针，2 个二级间接指针，1 个三级间接指针。假设块大小为 4KB，每个指针长度为 4B，每个目录项为 16B。试回答如下问题：

- (1) 仅使用直接指针，能索引的最大文件大小是多少？
- (2) 某个目录文件的 inode 只有两个有效的直接指针，那么该层目录下最多有多少个文件或目录？
- (3) 该文件系统支持的最大文件大小是多少？（可以用 K/M/G/TB 表示，取 2 位有效数字）

第三部分：《体系结构》

第九题，问答题，共 15 分

请给出 32 位二进制数分别视作无符号整数、整数原码、整数补码所表示的数的范围。

第十题 问答题，共 15 分

某处理器的高速缓冲存储器（Cache）容量为 256KB，每个 Cache 行大小为 32B。主存地址的宽度为 48 位，Cache 结构采用 8 路组相联，请回答如下问题：

- (1) 给出主存地址字段中各段的位数；
- (2) 假设 Cache 初始时空，CPU 依次从主存地址 0,1,2,...,199，读出 200 个字节（一次读出 1 个字节），并重复此次序读 5 遍，问命中率是多少？
- (3) 针对前述访问序列，若 Cache 访问速度是主存访问速度的 10 倍，试问有 Cache 和无 Cache 相比，速度提高了多少倍？

第十一题 问答题，共 15 分

简述 RISC 和 CISC 的特征，并对 RISC 和 CISC 的优缺点进行比较。

第十二题 问答题，共 15 分

简述经典五级静态流水线的组织方式，指出流水线指令相关的类型及其特点。

第四部分：《编译原理》

第十三题，问答题，共 15 分：

对正则式 $(01)^*1^+0(0|1)^*$

- (1) 画出识别该正则式的 DFA（12 分）
- (2) 你画出的 DFA 是最简的 DFA 吗？（3 分）

第十四题，问答题，共 15 分：

对于构建一个无二义性的上下文无关文法的预测分析器，回答如下问题：

- (1) 可以利用文法符号的 FIRST 集合和 FOLLOW 集合来判定该文法是否存在一个 LL(1)分析表，具体的判定条件是什么（6 分）？
- (2) 如果上述文法中不存在空串产生式，可以只用 FIRST 集合来判定该文法是否存在一个 LL(1)分析表吗（4 分）？

(3) 如果采用短语层次的错误恢复策略, 可以改变或增加分析栈中的文法符号吗? (3分) 至少说明一个理由 (2分)。

第十五题, 问答题, 共 15 分

现在的很多芯片都支持乱序执行, 可以对指令序列进行动态的指令调度以改善程序执行的效率。在这个背景下, 编译系统的指令调度优化还有意义吗 (5分)? 举出至少两个理由支持你的观点 (10分)。

第十六题, 问答题, 共 15 分

下述文法描述了一个简单的程序, 包括声明部分和赋值语句序列, 不包括控制流语句等成分。

```
P → D ; S
D → D Decl | Decl
Decl → Type namelist;
Type → int | float
Namelist → Namelist, id | id
S → S ; A | A
A → Lexpr := Rexpr
Lexpr → id
Rexpr → E
E → E + E | id
```

请回答如下问题:

- (1) 给出一个翻译方案实现的如下检查: (11分)
 - a) 在符号表中, 为每个变量标注其类型信息;
 - b) 每个变量一定要先声明后使用, 且不能重复声明;
 - c) 所有的变量, 先赋值再引用。例如, 程序中声明了变量 a, 但是在赋值号右侧的表达式使用 a 之前, 一定要已经完成了对 a 的赋值。

在给出的翻译方案中可以利用如下的数据结构和函数:

- a) id 的具体值由词法分析器给出, 可以由 id.lex 得到;
- b) 每个不同名字的 id 各自对应了一个符号表表项, 每个表项中用 ass 域表示变量是否已经被赋值, 其默认初值为 0, 表示未赋值;
- c) 可用 get_st (id.lex) 查询符号表, 如果该变量已经被声明, 则返回指向相应符号表的指针, 如果该变量尚未被声明, 则返回空指针;
- d) 可用 add_st (id.lex, type) 实现将一个 id 对应的表项加入到符号表中, 并填充类型信息;
- e) 可用 error () 报出错信息, 其参数是具体的错误信息, 不需报告错误位置。

- (2) 按照你给出的翻译方案, 是否可能至少报告一次错误? 为什么? (4分)