

2017 年全国硕士研究生入学统一考试

计算机科学与技术学科联考计算机学科专业基础综合试题

一、单项选择题：第 1~40 小题，每小题 2 分，共 80 分。下列每题给出的四个选项中，只有一个选项最符合试题要求。

1. 下列函数的时间复杂度是_____。

```
int func(int n){
    int i=0, sum=0;
    while(sum < n) sum += ++i;
    return i;
}
```

A. $O(\log n)$ B. $O(n^{1/2})$ C. $O(n)$ D. $O(n \log n)$

2. 下列关于栈的叙述中，错误的是_____。

- I. 采用非递归方式重写递归程序时必须使用栈
- II. 函数调用时，系统要用栈保存必要的信息
- III. 只要确定了入栈次序，即可确定出栈次序
- IV. 栈是一种受限的线性表，允许在其两端进行操作

A. 仅 I B. 仅 I、II、III
C. 仅 I、III、IV D. 仅 II、III、IV

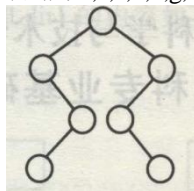
3. 适用于压缩存储稀疏矩阵的两种存储结构是_____。

A. 三元组表和十字链表 B. 三元组表和邻接矩阵
C. 十字链表和二叉链表 D. 邻接矩阵和十字链表

4. 要使一棵非空二叉树的先序序列与中序序列相同，其所有非叶结点须满足的条件是_____。

A. 只有左子树 B. 只有右子树
C. 结点的度均为 1 D. 结点的度均为 2

5. 已知一棵二叉树的树形如下图所示，其后序序列为 e,a,c,b,d,g,f，树中与结点 a 同层的结点是_____。



A. c B. d C. f D. g

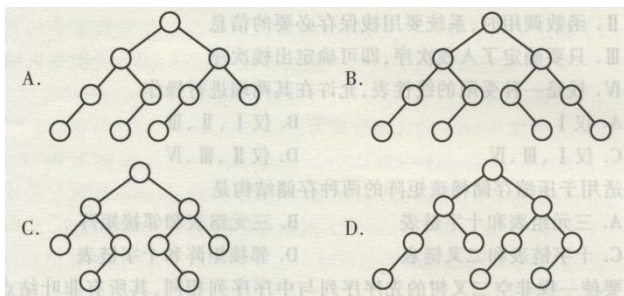
6. 已知字符集{a,b,c,d,e,f,g,h}，若各字符的哈夫曼编码依次是 0100, 10, 0000, 0101, 001, 011, 11, 0001，则编码序列 0100011001001011110101 的译码结果是_____。

A. acgabfh B. adbaggbb
C. afbeagd D. afeefgd

7. 已知无向图 G 含有 16 条边，其中度为 4 的顶点个数为 3，度为 3 的顶点个数为 4，其他顶点的度均小于 3。图 G 所含的顶点个数至少是_____。

A. 10 B. 11 C. 13 D. 15

8. 下列二叉树中，可能成为折半查找判定树（不含外部结点）的是_____。



9. 下列应用中, 适合使用 B+树的是_____。

- A. 编译器中的词法分析 B. 关系数据库系统中的索引
C. 网络中的路由表快速查找 D. 操作系统的磁盘空闲块管理

10. 在内部排序时, 若选择了归并排序而没有选择插入排序, 则可能的理由是_____。

- I. 归并排序的程序代码更短
II. 归并排序的占用空间更少
III. 归并排序的运行效率更高

- A. 仅 II B. 仅 III C. 仅 I、II D. 仅 I、III

11. 下列排序方法中, 若将顺序存储更换为链式存储, 则算法的时间效率会降低的是_____。

- I. 插入排序 II. 选择排序 III. 起泡排序 IV. 希尔排序 V. 堆排序

- A. 仅 I、II B. 仅 II、III C. 仅 III、IV D. 仅 IV、V

12. 假定计算机 M1 和 M2 具有相同的指令集体系结构 (ISA), 主频分别为 1.5GHz 和 1.2 GHz。在 M1 和 M2 上运行某基准程序 P, 平均 CPI 分别为 2 和 1, 则程序 P 在 M1 和 M2 上运行时间的比值是_____。

- A. 0.4 B. 0.625 C. 1.6 D. 2.5

13. 某计算机主存按字节编址, 由 4 个 64M×8 位的 DRAM 芯片采用交叉编址方式构成, 并与宽度为 32 位的存储器总线相连, 主存每次最多读写 32 位数据。若 double 型变量 x 的主存地址为 804 001AH, 则读取 x 需要的存储周期数是_____。

- A. 1 B. 2 C. 3 D. 4

14. 某 C 语言程序段如下:

```
for(i=0; i<=9; i++){
    temp = 1;
    for(j=0; j<=i; j++) temp *= a[j];
    sum += temp;
}
```

下列关于数组 a 的访问局部性的描述中, 正确的是_____。

- A. 时间局部性和空间局部性皆有
B. 无时间局部性, 有空间局部性
C. 有时间局部性, 无空间局部性
D. 时间局部性和空间局部性皆无

15. 下列寻址方式中, 最适合按下标顺序访问一维数组元素的是_____。

- A. 相对寻址 B. 寄存器寻址 C. 直接寻址 D. 变址寻址

16. 某计算机按字节编址, 指令字长固定且只有两种指令格式, 其中三地址指令 29 条, 二地址指令 107 条, 每个地址字段为 6 位, 则指令字长至少应该是_____。

- A. 24 位 B. 26 位 C. 28 位 D. 32 位

17. 下列关于超标量流水线特性的叙述中, 正确的是_____。

- I. 能缩短流水线功能段的处理时间
II. 能在一个时钟周期内同时发射多条指令
III. 能结合动态调度技术提高指令执行并行性

- A. 仅 II B. 仅 I、III C. 仅 II、III D. I、II 和 III

18. 下列关于主存储器 (MM) 和控制存储器 (CS) 的叙述中, 错误的是_____。

- A. MM 在 CPU 外, CS 在 CPU 内
 B. MM 按地址访问, CS 按内容访问
 C. MM 存储指令和数据, CS 存储微指令
 D. MM 用 RAM 和 ROM 实现, CS 用 ROM 实现

19. 下列关于指令流水线数据通路的叙述中, 错误的是_____。

- A. 包含生成控制信号的控制部件
 B. 包含算术逻辑运算部件 (ALU)
 C. 包含通用寄存器组和取指部件
 D. 由组合逻辑电路和时序逻辑电路组合而成

20. 下列关于多总线结构的叙述中, 错误的是_____。

- A. 靠近 CPU 的总线速度较快
 B. 存储器总线可支持突发传送方式
 C. 总线之间须通过桥接器相连
 D. PC I-Express×16 采用并行传输方式

21. I/O 指令实现的数据传送通常发生在_____。

- A. I/O 设备和 I/O 端口之间 B. 通用寄存器和 I/O 设备之间
 C. I/O 端口和 I/O 端口之间 D. 通用寄存器和 I/O 端口之间

22. 下列关于多重中断系统的叙述中, 错误的是_____。

- A. 在一条指令执行结束时响应中断
 B. 中断处理期间 CPU 处于关中断状态
 C. 中断请求的产生与当前指令的执行无关
 D. CPU 通过采样中断请求信号检测中断请求

23. 假设 4 个作业到达系统的时刻和运行时间如下表所示。

作业	到达时刻 t	运行时间
J1	0	3
J2	1	3
J3	1	2
J4	3	1

系统在 t=2 时开始作业调度。若分别采用先来先服务和短作业优先调度算法, 则选中的作业分别是_____。

- A. J2、J3 B. J1、J4 C. J2、J4 D. J1、J3

24. 执行系统调用的过程包括如下主要操作:

- ①返回用户态 ②执行陷入(trap)指令
 ③传递系统调用参数 ④执行相应的服务程序

正确的执行顺序是_____。

- A. ②->③->①->④ B. ②->④->③->①
 C. ③->②->④->① D. ③->④->②->①

25. 某计算机按字节编址, 其动态分区内存管理采用最佳适应算法, 每次分配和回收内存后都对空闲分区链重新排序。当前空闲分区信息如下表所示。

分区起始地址	20 K	500 K	1000 K	200 K
分区大小	40 KB	80 KB	100 KB	200 KB

回收起始地址为 60 K、大小为 140 KB 的分区后, 系统中空闲分区的数量、空闲分区链第一个分区的起始地址和大小分别是_____。

- A. 3、20 K、380 KB B. 3、500 K、80 KB
 C. 4、20 K、180 KB D. 4、500 K、80 KB

26. 某文件系统的簇和磁盘扇区大小分别为 1 KB 和 512 B。若一个文件的大小为 1 026B, 则系统分配给该文件的磁盘空间大小是_____。

- A. 1026 B B. 1536 B C. 1538 B D. 2048 B

27. 下列有关基于时间片的进程调度的叙述中, 错误的是_____。

- A. 时间片越短, 进程切换的次数越多, 系统也越大
B. 当前进程的时间片用完后, 该进程状态由执行态变为阻塞态
C. 时钟中断发生后, 系统会修改当前进程在时间片内的剩余时间
D. 影响时间片大小的主要因素包括响应时间、系统开销和进程数量等

28. 与单道程序系统相比, 多道程序系统的优点是_____。

- I. CPU 利用率高 II. 系统开销小
III. 系统吞吐量高 IV. I/O 设备利用率高
A. 仅 I、III B. 仅 I、IV
C. 仅 II、III D. 仅 I、III、IV

29. 下列选项中, 磁盘逻辑格式化程序所做的工作是_____。

- I. 对磁盘进行分区
II. 建立文件系统的根目录
III. 确定磁盘扇区校验码所占位数
IV. 对保存空闲磁盘块信息的数据结构进行初始化
A. 仅 II B. 仅 II、IV
C. 仅 III、IV D. 仅 I、II、IV

30. 某文件系统中, 针对每个文件, 用户类别分为 4 类: 安全管理员、文件主、文件主的伙伴、其他用户; 访问权限分为 5 种: 完全控制、执行、修改、读取、写入。若文件控制块中用二进制位串表示文件权限, 为表示不同类别用户对一个文件的访问权限, 则描述文件权限的位数至少应为_____。

- A. 5 B. 9 C. 12 D. 20

31. 若文件 f1 的硬链接为 f2, 两个进程分别打开 f1 和 f2, 获得对应的文件描述符为 fd1 和 fd2, 则下列叙述中, 正确的是_____。

- I. f1 和 f2 的读写指针位置保持相同
II. f1 和 f2 共享同一个内存索引结点
III. fd1 和 fd2 分别指向各自的用户打开文件表中的一项
A. 仅 I B. 仅 II、III C. 仅 I、II D. I、II 和 III

32. 系统将数据从磁盘读到内存的过程包括以下操作:

- ①DMA 控制器发出中断请求
②初始化 DMA 控制器并启动磁盘
③从磁盘传输一块数据到内存缓冲区
④执行“DMA 结束”中断服务程序

正确的执行顺序是_____。

- A. ③→①→②→④ B. ②→③→①→④
C. ②→①→③→④ D. ①→②→④→③

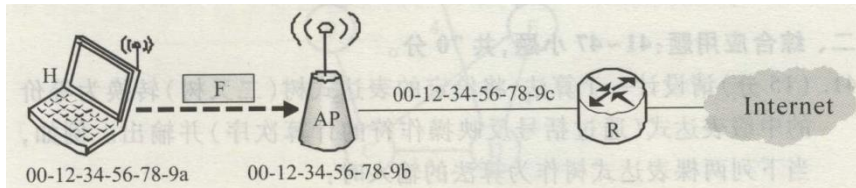
33. 假设 OSI 参考模型的应用层欲发送 400 B 的数据 (无拆分), 除物理层和应用层之外, 其他各层在封装 PDU 时均引入 20 B 的额外开销, 则应用层数据传输效率约为_____。

- A. 80% B. 83% C. 87% D. 91%

34. 若信道在无噪声情况下的极限数据传输速率不小于信噪比为 30dB 条件下的极限数据传输速率, 则信号状态数至少是_____。

- A. 4 B. 8 C. 16 D. 32

35. 在下图所示的网络中, 若主机 H 发送一个封装访问 Internet 的 IP 分组的 IEEE 802.11 数据帧 F, 则帧 F 的地址 1、地址 2 和地址 3 分别是_____。



- A. 00-12-34-56-78-9a, 00-12-34-56-78-9b, 00-12-34-56-78-9c
 B. 00-12-34-56-78-9b, 00-12-34-56-78-9a, 00-12-34-56-78-9c
 C. 00-12-34-56-78-9b, 00-12-34-56-78-9c, 00-12-34-56-78-9a
 D. 00-12-34-56-78-9a, 00-12-34-56-78-9c, 00-12-34-56-78-9b

36. 下列 IP 地址中, 只能作为 IP 分组的源 IP 地址但不能作为目的 IP 地址的是_____。

- A. 0.0.0.0
 B. 127.0.0.1
 C. 200.10.10.3
 D. 255.255.255.255

37. 直接封装 RIP、OSPF、BGP 报文的协议分别是_____。

- A. TCP、UDP、IP
 B. TCP、IP、UDP
 C. UDP、TCP、IP
 D. UDP、IP、TCP

38. 若将网络 21.3.0.0/16 划分为 128 个规模相同的子网, 则每个子网可分配的最大 IP 地址个数是_____。

- A. 254
 B. 256
 C. 510
 D. 512

39. 若甲向乙发起一个 TCP 连接, 最大段长 MSS=1 KB, RTT=5 ms, 乙开辟的接收缓存为 64 KB, 则甲从连接建立成功至发送窗口达到 32 KB, 需经过的时间至少是_____。

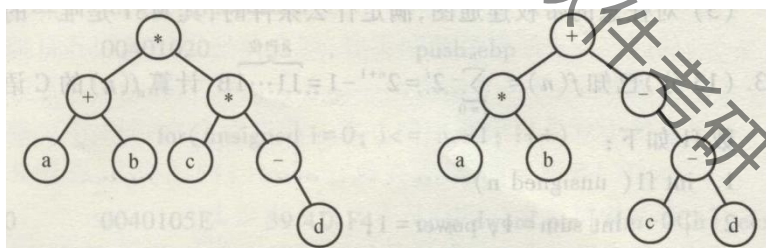
- A. 25 ms
 B. 30 ms
 C. 160 ms
 D. 165 ms

40. 下列关于 FTP 协议的叙述中, 错误的是_____。

- A. 数据连接在每次数据传输完毕后就关闭
 B. 控制连接在整个会话期间保持打开状态
 C. 服务器与客户端的 TCP 20 端口建立数据连接
 D. 客户端与服务器的 TCP 21 端口建立控制连接

二、综合应用题: 第 41~47 小题, 共 70 分。

41. (15 分) 请设计一个算法, 将给定的表达式树(二叉树)转换为等价的中缀表达式(通过括号反映操作符的计算次序)并输出。例如, 当下列两棵表达式树作为算法的输入时:



输出的等价中缀表达式分别为 $(a+b)*(c*(-d))$ 和 $(a*b)+(-(c-d))$ 。

二叉树结点定义如下:

```
typedef struct node{
    char data[10];           //存储操作数或操作符
    struct node *left, *right;
}BTree;
```

要求:

(1)给出算法的基本设计思想。

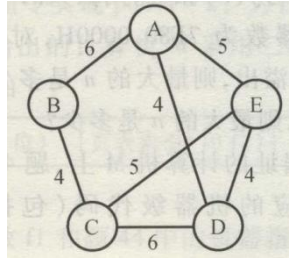
(2)根据设计思想, 采用 C 或 C++ 语言描述算法, 关键之处给出注释。

42. (8 分) 使用 Prim(普里姆)算法求带权连通图的最小(代价)生成树(MST)。请回答下列问题。

(1)对下列图 G, 从顶点 A 开始求 G 的 MST, 依次给出按算法选出的边。

(2)图 G 的 MST 是唯一的吗?

(3)对任意的带权连通图，满足什么条件时，其 MST 是唯一的？



$$f(n) = \sum_{i=0}^n 2^i = 2^{n+1} - 1 = \overbrace{11 \cdots 1}^{n+1 \text{ 位}} \text{B}$$

43.(13 分) 已知 $f(n) = \sum_{i=0}^n 2^i = 2^{n+1} - 1 = \overbrace{11 \cdots 1}^{n+1 \text{ 位}} \text{B}$ ，计算 $f(n)$ 的 C 语言函数 $f1$ 如下：

```
int f1(unsigned n){
    int sum=1, power=1;
    for(unsigned i=0;i<=n-1;i++){
        power *= 2;
        sum += power;
    }
    return sum;
}
```

将 $f1$ 中的 int 都改为 $float$ ，可得到计算 $f(n)$ 的另一个函数 $f2$ 。假设 $unsigned$ 和 int 型数据都占 32 位， $float$ 采用 IEEE 754 单精度标准。请回答下列问题。

(1)当 $n=0$ 时， $f1$ 会出现死循环，为什么？若将 $f1$ 中的变量 i 和 n 都定义为 int 型，则 $f1$ 是否还会出现死循环？为什么？

(2) $f1(23)$ 和 $f2(23)$ 的返回值是否相等？机器数各是什么（用十六进制表示）？

(3) $f1(24)$ 和 $f2(24)$ 的返回值分别为 33 554 431 和 33 554 432.0，为什么不相等？

(4) $f(31)=2^{32}-1$ ，而 $f1(31)$ 的返回值却为 -1，为什么？若使 $f1(n)$ 的返回值与 $f(n)$ 相等，则最大的 n 是多少？

(5) $f2(127)$ 的机器数为 7F80 0000H，对应的值是什么？若使 $f2(n)$ 的结果不溢出，则最大的 n 是多少？若使 $f2(n)$ 的结果精确（无舍入），则最大的 n 是多少？

44.(10 分) 在按字节编址的计算机 M 上，题 43 中 $f1$ 的部分源程序(阴影部分)与对应的机器级代码（包括指令的虚拟地址）如下：

		int f1(unsigned n)	
1	00401020	55	push ebp

		for(unsigned i=0; i<= n -1; i++)	

20	0040105E	39 4D F4	cmp dword ptr [ebp-0Ch],ecx

		{ power *= 2;	

23	00401066	D1 E2	shl edx,1

		return sum;	

35	0040107F	C3	ret

其中，机器级代码行包括行号、虚拟地址、机器指令和汇编指令。请回答下列问题。

(1)计算机 M 是 RISC 还是 CISC？为什么？

(2)f1 的机器指令代码共占多少字节？要求给出计算过程。

(3)第 20 条指令 `cmp` 通过 `i` 减 `n-1` 实现对 `i` 和 `n-1` 的比较。执行 `f1(0)` 过程中当 `i=0` 时，`cmp` 指令执行后，进/借位标志 `CF` 的内容是什么？要求给出计算过程。

(4)第 23 条指令 `shl` 通过左移操作实现了 `power * 2` 运算，在 `f2` 中能否也用 `shl` 指令实现 `power * 2`？为什么？

45. (7 分) 假定题 44 给出的计算机 `M` 采用二级分页虚拟存储管理方式，虚拟地址格式如下：

页目录号（10 位）	页表索引（10 位）	页内偏移量（12 位）
------------	------------	-------------

请针对题 43 的函数 `f1` 和题 44 中的机器指令代码，回答下列问题。

(1)函数 `f1` 的机器指令代码占多少页？

(2)取第 1 条指令（`push ebp`）时，若在进行地址变换的过程中需要访问内存中的页目录和页表，则会分别访问它们各自的第几个表项（编号从 0 开始）？

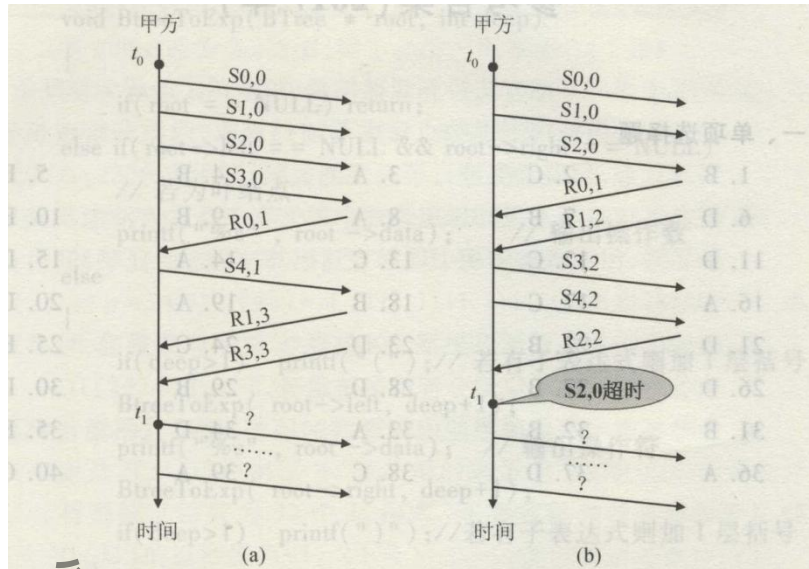
(3)`M` 的 I/O 采用中断控制方式。若进程 `P` 在调用 `f1` 之前通过 `scanf()` 获取 `n` 的值，则在执行 `scanf()` 的过程中，进程 `P` 的状态会如何变化？CPU 是否会进入内核态？

46. (8 分) 某进程中有 3 个并发执行的线程 `thread1`、`thread2` 和 `thread3`，其伪代码如下所示。

<pre>// 复数的结构类型定义 typedef struct { float a; float b; } cnum; cnum x, y, z; // 全局变量 // 计算两个复数之和 cnum add(cnum p, cnum q) { cnum s; s.a = p.a + q.a; s.b = p.b + q.b; return s; }</pre>	<pre>thread1 { cnum w; w = add(x, y); }</pre>	<pre>thread3 { cnum w; w.a = 1; w.b = 1; z = add(z, w); y = add(y, w); }</pre>
	<pre>thread2 { cnum w; w = add(y, z); }</pre>	

请添加必要的信号量和 P、V(或 `wait()`、`signal()`) 操作，要求确保线程互斥访问临界资源，并且最大程度地并发执行。

47. (9 分) 甲乙双方均采用后退 `N` 帧协议 (GBN) 进行持续的双向数据传输，且双方始终采用捎带确认，帧长均为 1000 B。 $S_{x,y}$ 和 $R_{x,y}$ 分别表示甲方和乙方发送的数据帧，其中：`x` 是发送序号；`y` 是确认序号（表示希望接收对方的下一帧序号）；数据帧的发送序号和确认序号字段均为 3 比特。信道传输速率为 100Mbps， $RTT=0.96ms$ 。下图给出了甲方发送数据帧和接收数据帧的两种场景，其中 t_0 为初始时刻，此时甲方的发送和确认序号均为 0， t_1 时刻甲方有足够多的数据待发送。



请回答下列问题。

- (1)对于图(a), t_0 时刻到 t_1 时刻期间, 甲方可以断定乙方已正确接收的数据帧数是多少? 正确接收的是哪几个帧(请用 Sx,y 形式给出)?
- (2)对于图(a), 从 t_1 时刻起, 甲方在不出现超时且未收到乙方新的数据帧之前, 最多还可以发送多少个数据帧? 其中第一个帧和最后一个帧分别是哪个(请用 Sx,y 形式给出)?
- (3)对于图(b), 从 t_1 时刻起, 甲方在不出现新的超时且未收到乙方新的数据帧之前, 需要重发多少个数据帧? 重发的第一个帧是哪个(请用 Sx,y 形式给出)?
- (4)甲方可以达到的最大信道利用率是多少?

2017 年计算机学科专业基础综合试题参考答案

一、单项选择题

- | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|
| 1. B | 2. C | 3. A | 4. B | 5. B | 6. D | 7. B | 8. A |
| 9. B | 10. B | 11. D | 12. C | 13. C | 14. A | 15. D | 16. A |
| 17. C | 18. B | 19. A | 20. D | 21. D | 22. B | 23. D | 24. C |
| 25. B | 26. D | 27. B | 28. D | 29. B | 30. D | 31. B | 32. B |
| 33. A | 34. D | 35. B | 36. A | 37. D | 38. C | 39. A | 40. C |

二、综合应用题

41. 解析:

(1)算法的基本设计思想

表达式树的中序序列加上必要的括号即为等价的中缀表达式。可以基于二叉树的中序遍历策略得到所需的表达式。(3分)

表达式树中分支结点所对应的子表达式的计算次序, 由该分支结点所处的位置决定。为得到正确的中缀表达式, 需要在生成遍历序列的同时, 在适当位置增加必要的括号。显然, 表达式的最外层(对应根结点)及操作数(对应叶结点)不需要添加括号。(2分)

(2)算法实现(10分)

将二叉树的中序遍历递归算法稍加改造即可得本题答案。除根结点和叶结点外, 遍历到其他结点时在遍历其左子树之前加上左括号, 在遍历完右子树后加上右括号。

```
void BtreeToE(BTree *root){
```



```

    BtreeToExp(root, 1);                //根的高度为 1
}
void BtreeToExp( BTree *root, int deep)
{
    if(root == NULL) return;           //空结点返回
    else if(root->left==NULL&&root->right==NULL) //若为叶结点
        printf("%s", root->data); //输出操作数, 不加括号
    else{
        if(deep>1) printf("(");        //若有子表达式则加 1 层括号
        BtreeToExp(root->left, deep+1);
        printf("%s", root->data); //输出操作符
        BtreeToExp(root->right, deep+1);
        if(deep>1) printf(")");        //若有子表达式则加 1 层括号
    }
}
}

```

【评分说明】①若考生设计的算法满足题目的功能要求, 则 (1)、(2) 根据所实现算法的策略及输出结果给分, 细则见下表。

分数	备注
15	采用中序遍历算法且正确, 括号嵌套正确, 层数适当。
14	采用中序遍历算法且正确, 括号嵌套正确, 但括号嵌套层数 过多。例如, 表达式最外层加上括号, 或操作数加括号如 (a)。
11	采用中序遍历算法, 但括号嵌套层数不完全正确。例如, 左 右括号数量不匹配。
9	采用中序遍历算法, 但没有考虑括号。
≤7	其他

②若考生采用其他方法得到正确结果, 可参照①的评分标准给分。

③如果程序中使用了求结点深度等辅助函数, 但没有给出相应的实现过程, 只要考生进行了必要的说明, 可不扣分。

④若在算法的基本设计思想描述中因文字表达没有清晰反映出算法思路, 但在算法实现中能够表达出算法思想且正确的, 可参照①的标准给分。

⑤若算法的基本设计思想描述或算法实现中部分正确, 可参照①中各种情况的相应给分标准酌情给分。

⑥参考答案中只给出了使用 C 语言的版本, 使用 C++语言的答案参照以上评分标准。

42. 解析:

(1)Prim 算法属于贪心策略。算法从一个任意的顶点开始, 一直长大到覆盖图中所有顶点为止。算法每一步在连接树集合 S 中顶点和其他顶点的边中, 选择一条使得树的总权重增加最小的边加入集合 S。当算法终止时, S 就是最小生成树。

①S 中顶点为 A, 候选边为(A,D)、(A,B)、(A,E), 选择(A,D)加入 S。

②S 中顶点为 A、D, 候选边为(A,B)、(A,E)、(D,E)、(C,D), 选择(D,E), 加入 S。

③S 中顶点为 A、D、E, 候选边为(A,B)、(C,D)、(C,E), 选择(C,E)加入 S。

④S 中顶点为 A、D、E、C, 候选边为(A,B)、(B,C), 选择(B,C)加入 S。

⑤S 就是最小生成树。

依次选出的边为:

(A, D),(D, E),(C, E),(B, C) (4 分)

【评分说明】每正确选对一条边且次序正确, 给 1 分。若考生选择的边正确, 但次序不完全正确, 酌情给分。

(2)图 G 的 MST 是唯一的。(2 分) 第一小题的最小生成树包括了图中权值最小的四条边, 其他边都比这四条边大, 所以此图 MST 唯一。

(3)当带权连通图的任意一个环中所包含的边的权值均不相同, 其 MST 是唯一的。(2 分) 此题不要求回答充分必要条件, 所以回答一个限制边权值的充分条件即可。

【评分说明】①若考生答案中给出的是其他充分条件，例如“带权连通图的所有边的权值均不相同”，同样给分。

②若考生给出的充分条件对图的顶点数和边数做了某些限制,例如,限制了图中顶点的个数(顶点个数少于3个)、限制了图的形状(图中没有环)等,则最高给1分。

③答案部分正确，酌情给分。

43. 解析:

(1) 由于 i 和 n 是 unsigned 型, 故 “ $i \leq n-1$ ” 是无符号数比较, $n=0$ 时, $n-1$ 的机器数为全 1, 值是 $2^{32}-1$, 为 unsigned 型可表示的最大数, 条件 “ $i \leq n-1$ ” 永真, 因此出现死循环。(2 分)

若 i 和 n 改为 int 类型，则不会出现死循环。（1 分）

因为“ $i \leq n-1$ ”是带符号整数比较, $n=0$ 时, $n-1$ 的值是 -1, 当 $i=0$ 时条件“ $i \leq n-1$ ”不成立, 此时退出 for 循环。
(1 分)

(2) f1(23)与f2(23)的返回值相等。(1分) $f(23)=2^{23+1}-1=2^{24}-1$ ，它的二进制形式是24个1。int占32位，没有溢出。float有1个符号位，8个指数位，23个底数位，23个底数位可以表示24位的底数。所以两者返回值相等。

f1(23)的机器数是 00FF FFFFH, (1 分)

f2(23)的机器数是 4B7F FFFFH。 (1 分)

显而易见前者是 24 个 1, 即 0000 0000 1111 1111 1111 1111 1111 1111₍₂₎, 后者符号位是 0, 指数位为 $23+127_{(10)}=1001\ 0110_{(2)}$, 底数位是 111 1111 1111 1111 1111 1111₍₂₎。

(3) 当 $n=24$ 时, $f(24)=1\ 1111\ 1111\ 1111\ 1111\ 1111\ 1111\ B$, 而 `float` 型数只有 24 位有效位, 舍入后数值增大, 所以 $f2(24)$ 比 $f1(24)$ 大 1。(1 分)

【评分说明】只要说明 f2(24) 需舍人处理即可给分。

(4) 显然 f(31) 已超出了 int 型数据的表示范围, 用 f1(31) 实现时得到的机器数为 32 个 1, 作为 int 型数解释时其值为 -1, 即 f1(31) 的返回值为 -1。(1 分)

因为 int 型最大可表示数是 0 后面加 31 个 1，故使 f1(n) 的返回值与 f(n) 相等的最大 n 值是 30。（1 分）

【评分说明】对于第二问，只要给出 $n=30$ 即可给分。

(5) IEEE 754 标准用“阶码全 1、尾数全 0”表示无穷大。f2 返回值为 float 型，机器数 7F80 0000H 对应的值是+ ∞ 。(1 分)

当 $n=126$ 时, $f(126)=2^{127}-1=1.1\cdots 1\times 2^{126}$, 对应阶码为 $127+126=253$, 尾数部分舍入后阶码加 1, 最终阶码为 254, 是 IEEE 754 单精度格式表示的最大阶码。故使 $f2$ 结果不溢出的最大 n 值为 126。(1 分)

当 $n=23$ 时, $f(23)$ 为 24 位 1, float 型数有 24 位有效位, 所以不需舍入, 结果精确。故使 $f2$ 获得精确结果的最大 n 值为 23。(1 分)

【评分说明】对于第二问，只要给出 $n=23$ ，即可给分。对于第三问，只要给出 $n=126$ ，即可给分。

44. 解析:

(1)M 为 CISC。(1 分)

M 的指令长短不一，不符合 RISC 指令系统特点。（1 分）

(2) f1 的机器代码占 96B。 (1 分)

因为 f1 的第一条指令“push ebp”所在的虚拟地址为 0040 1020H，最后一条指令“ret”所在的虚拟地址为 0040 107FH，所以，f1 的机器指令代码长度为 $0040\ 107FH - 0040\ 1020H + 1 = 60H = 96$ 个字节。（1 分）

(3) $CF=1$ 。(1 分)

cmp 指令实现 i 与 n-1 的比较功能, 进行的是减法运算。在执行 f1(0) 过程中, n=0, 当 i=0 时, i=0000 0000H, 并且 n-1=FFFF FFFFH。因此, 当执行第 20 条指令时, 在补码加/减运算器中执行“0 减 FFFF FFFFH”的操作, 即 0000 0000H + 0000 0000H+1 = 0000 0001H, 此时, 进位输出 C = 0, 减法运算时的借位标志 CF = C ⊕ 1 = 1。(2 分)

(4) f2 中不能用 shl 指令实现 power*2。 (1 分)

因为 shl 指令用来将一个整数的所有有效数位作为一个整体左移; 而 f2 中的变量 power 是 float 型, 其机器数中不包含最高有效数位, 但包含了阶码部分, 将其作为一个整体左移时并不能实现“乘 2”的功能, 因而 f2 中不能用 shl 指令实现 power*2。(2 分) 浮点数运算比整型运算要复杂, 耗时也较长。

45. 解析:

(1) 函数 f1 的代码段中所有指令的虚拟地址的高 20 位相同，因此 f1 的机器指令代码在同一页中，仅占用 1 页。
(1 分) 页目录号用于寻找页目录的表项，该表项包含页表的位置。页表索引用于寻找页表的表项，该表项包含页的位置。

(2) push ebp 指令的虚拟地址的最高 10 位（页目录号）为 00 0000 0001，中间 10 位（页表索引）为 00 0000 0001，所以，取该指令时访问了页目录的第 1 个表项，（1 分）在对应的页表中访问了第 1 个表项。（1 分）

(3) 在执行 scanf() 的过程中，进程 P 因等待输入而从执行态变为阻塞态。（1 分）输入结束时，P 被中断处理程序唤醒，变为就绪态。（1 分）P 被调度程序调度，变为运行态。（1 分）CPU 状态会从用户态变为内核态。（1 分）

46. 解析：

先找出线程对在各个变量上的互斥、并发关系。如果是一读一写或两个都是写，那么这就是互斥关系。每一个互斥关系都需要一个信号量进行调节。

```
semaphore mutex_y1=1; //mutex_y1 用于 thread1 与 thread3 对变量 y 的互斥访问。
semaphore mutex_y2=1; //mutex_y2 用于 thread2 与 thread3 对变量 y 的互斥访问。
semaphore mutex_z=1; //mutex_z 用于变量 z 的互斥访问。
```

互斥代码如下：（5 分）

thread1	thread2	thread3
<pre> enum w; wait(mutex_y1); w = add(x, y); signal(mutex_y1); </pre>	<pre> enum w; wait(mutex_y2); wait(mutex_z); w = add(x, z); signal(mutex_z); signal(mutex_y2); </pre>	<pre> enum w; w.a = 1; w.b = 1; wait(mutex_z); z = add(z, w); signal(mutex_z); wait(mutex_y1); wait(mutex_y2); y = add(y, w); signal(mutex_y1); signal(mutex_y2); </pre>

【评分说明】

①各线程与变量之间的互斥、并发情况及相应评分见下表。

线程对 变量	thread1 和 thread2	thread2 和 thread3	thread1 和 thread3	给分
x	不共享	不共享	不共享	1 分
y	同时读	读写互斥	读写互斥	3 分
z	不共享	读写互斥	不共享	1 分

②若考生仅使用一个互斥信号量，互斥代码部分的得分最多给 2 分。

③答案部分正确，酌情给分。

47. 解析：

(1) t_0 时刻到 t_1 时刻期间，甲方可以断定乙方已正确接收了 3 个数据帧，（1 分）分别是 S0, 0、S1, 0、S2, 0。（1 分）R3, 3 说明乙发送的数据帧确认号是 3，即希望甲发送序号 3 的数据帧，说明乙已经接收了序号为 0-2 的数据帧。

(2) 从 t_1 时刻起，甲方最多还可以发送 5 个数据帧，（1 分）其中第一个帧是 S5, 2，（1 分）最后一个数据帧是 S1, 2。（1 分）发送序号 3 位，有 8 个序号。在 GBN 协议中，序号个数 = 发送窗口 + 1，所以这里发送窗口最大为 7。此时已发送了 S3, 0 和 S4, 1，所以最多还可以发送 5 个帧。

(3) 甲方需要重发 3 个数据帧，（1 分）重发的第一个帧是 S2, 3。（1 分）在 GBN 协议中，接收方发送了 N 帧后，检测出错，则需要发送出错帧及其之后的帧。S2, 0 超时，所以重发的第一帧是 S2。已收到乙的 R2 帧，所以确认号应为 3。

(4) 甲方可以达到的最大信道利用率是：

$$\frac{7 \times \frac{8 \times 1000}{100 \times 10^6}}{0.96 \times 10^{-3} + 2 \times \frac{8 \times 1000}{100 \times 10^6}} \times 100\% = 50\% \quad (2 \text{ 分})$$

$U = \text{发送数据的时间} / \text{从开始发送第一帧到收到第一个确认帧的时间} = N * T_d / (T_d + RTT + T_a)$

U 是信道利用率，N 是发送窗口的最大值，T_d 是发送一数据帧的时间，RTT 是往返时间，T_a 是发送一确认帧的时间。这里采用捎带确认，T_d=T_a。

【评分说明】答案部分正确，酌情给分。

微信公众

计算机/软件工程专业

每个学校的

考研真题/复试资料/考研经验

考研资讯/报录比/分数线

免费分享



微信 扫一扫

关注微信公众号

计算机与软件考研