

## 2018 浙江大学 878 专业课试题回忆版

写在前面:

如果有错误, 那就我也没招了。

以下提供了一些不负责并且可能不靠谱的复习建议, 请理智阅读。

### 第一部分 程序设计语言 (C)

#### 一. 单选题 (一题 2 分, 15 题, 一共 30 分)

1. 下列不能正确表述  $x$  属于  $[-5, 0]$  的是

A.  $x \geq -5 \&\& x \leq 0$

B.  $x \leq 0 \&\& x \geq -5$

B.  $!(x < -5 \&\& x > 0)$

D.  $-5 \leq x \leq 0$

2. 下面选项中与 `while` 括号中等价的是

`while(x%3) {`

.....

`}`

A.  $x \% 3 \neq 0$    B.  $x \% 3 == 1$    C.  $x \% 3 == 2$    D.  $x \% 3 == 3$

后面的题记不起来了, 但是比较难的考点是二维数组的存储方式和两个下标怎么对应到一维线性的位置, 就是 `a[-1][3]` 这样的访问为何不越界。 `stdin` 和 `stdout` 的可读性和可写性, 还有 `static` 变量的特性, 但是这个题最后返回的是一个局部变量, 题目不难, 就是很考耐心, 一定要慢慢读题, 慢慢阅读代码。选择题和填空题最喜欢就是字符串数组 指针数组 数组指针 二维指针, 这个务必要看, 要看明白了, 今年 C 语言有 30+ 的题目考点在上面, 还有一个题是解析 `++p->c`

这个表示意思（`p` 是指向结构体数组第二位的指针，`c` 是结构体成员），还有就是指针和地址的差别，为什么两个字符串常量相减是错的（可以从字符串常量在 C 程序的存储位置来理解），有一题是“1234”[2] 这样的操作。剩下的没啥难题了应该。

## 二. 读程序（一题 6 分， 5 题， 一共 30 分）

第一个题的考点是将插入排序的两个循环的内层循环封装成一个函数。题目看起来吓人，自己脑子跑起来也比较容易出错。第二个题的考点是进制转换，16 进制转化成 10 进制，但是他不是真的做转换，会加入一些无关的信息，让你模拟起来也比较恶心。第三个的考点是 `char *s[] = {"computer", "science"}; char *p = s;` 然后 `p[1]++` 是什么东西。第四个考点是，用异或运算实现两个数的交换。这四个题都是不难但是写起来贼恶心人那种，所以千万要稳住，一步步来，不要因为题目简单就放松警惕。

第五个题是程序填空，就是要你把一个链表中的比如 1 2 6 3 7 这样的导致链表递增的元素删除掉，题目不是很难，就不要被吓到，稳住就行。

总结：C 部分的题不是特别难，就是考细节，C 里面有些鸡爪旮旯的考点都会涉及，比如 `stdin` 的读写特性这种，但是主要的还是考细心和对常见 C 语言考点的熟

练，我个人的不负责建议就是刷完浙大教材和浙大的习题以后，可以把近几年的浙大 C 期末考试题写一下，挺有出题的风格，读程序写结果的第一题一题在浙大的 11 年还是 15 年的期末就有出现过类似的，总的来说就是拿一个 80% 的分还是相对容易的，剩下 20% 就比较考验细节和细心的。二级题目没写，如果感觉上面的几个如果感觉训练度不够还是可以写一下的。如果有时间的话，不负责任的建议读一下 csapp 的第三章，里面涉及了 C 的各个主要部分在计算机底层的实现方式，考试不会考，但是能加深你对 C 的理解(大概也有帮助科学吹牛的作用)。

## 第二部分 数据结构（共 90 分）

### 一. 单选题（我也不记得多少题多少分了）

ds 的选择题也不是很困难，风格和王道差不多有考察了队列 堆栈 和 线性表的差别，还有比较反常今年吧 AVL 的建树，并且前序遍历，还有堆的插入然后把堆的内容选出来，还有就是给你一个图，把 dijkstra 算法的选点顺序找出来，对一个有向无环图进行深搜，然

后回溯的时候的节点是拓扑序 逆拓扑序 还是 无序 还是不确定。还有一题是给你一个序列第二趟排序的结果让你判断这个序列是用什么排序，也是比较新的题，王道的这一类题一般都是一趟排序。选择题难度和王道的选择差不多，没有明显比王道难或者明显比王道弱。难度大概参考历届真题的选择，今年比较特殊的是把 **avl** 和堆两个本来是大题必考的知识点放在了选择题上。不知道 **19** 的变化趋势是怎么样的。但是作为 **18** 的考生复习，王道+真题的复习是应该足以应付这个选择题的。**878** 数据结构出题的知识点还是相对有规律的。

## 二. 数据结构大题

第一题是给你后序遍历和中序遍历建树，让你输出前序遍历（**18** 分）。第二题是用线性探测法解决冲突的哈希表，让你填表，然后让你计算在第一次冲突发生时的填装因子（历届真题没有出现过）。第三题，程序填空，是关于 **Floyd** 算法的实现，这个是新题，真题都没有出现过。历届真题选择题也没出现过，所以考纲里面的算法都应当重视，尽量都能手动实现吧，最坑的是这个 **floyd** 在王道上面也只是说了原理，所以务必以考纲为纲，**408** 不重视的不代表 **878** 不会考。第

四题是算法实现题，在一下定义的树中找到两个节点  $u$  和  $v$  的最早公共祖先（就是  $u$  和  $v$  的所有公共祖先深度最大的点）

```
typedef Tree *node;
struct node{
    Int key;
    Int left;
    Int right;
}
```

总结：整张卷子都不难，你能做，但是不一定能拿很高的分数，不像数学，你看了也不会做，复习的时候要多训练对细节把握，因为 878 卷子的题目比较少，所有每个题的分值都很高，一错就是 6~8 分没了，因此细节决定成败啊。另外群文件里还有一个叫[聊天记录]的文件里面有今年四个比较难的题的题目，可以看一看。