

简答题(real:刷真题分类, base:刷基础分类, pro:刷提升分类)

1. 翻译程序有哪些工作方式? 他们有什么区别?

- 翻译程序有两种工作方式: 解释方式和编译方式
- 编译方式是将高级语言源程序通过编译程序翻译成机器语言目标代码;解释方式是对高级语言源程序进行逐句解释,解释一句就执行一句,但不产生机器语言目标代码

2. 简述程序设计语言的发展。

- 计算机语言已经经历了机器语言、汇编语言、高级语言三个阶段。
- 机器语言是由 0、1 二进制代码组成的能被机器直接理解、执行的指令集合,是计算机唯一能够直接执行的语言。
- 汇编语言采用一定的助记符来代替机器语言中的指令和数据,用汇编语言编制的程序必须经过汇编程序翻译成计算机所能识别的机器语言程序后,才能被计算机执行。
- 高级语言编写的程序易学、易读、易修改, 通用性好,不依赖于机器

3. real, 十进制数 2022 转化为二进制数是多少?转化为十六进制数是多少?

```
/*
十进制2022转换为二进制为 0000011001010110 => 11111100110
十进制2022转换为十六进制为
每4个进行拆分组合, 结果为0x656 => 0x7E6
```

```
十进制转二进制, 直接除以2倒序拿余数
十进制转16进制, 直接除以16倒序拿余数
*/
```

3. 将二进制 101101.01111 分别转换为十进制, 八进制和十六进制

(101101.0111) 的二进制= $1 \times 2^5 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^0 + 1 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} = 45.4375$

其中: $1 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} = 1/4 + 1/8 + 1/16 = 7/16 = 0.4375$

4. 什么是常量? 什么是变量?

常量: 在程序运行过程中其值不能改变的量

变量: 在程序运行过程中其值可以改变的量

5. 简述变量的命名规则。

- **变量的命名规则为, 第一位为字母或下划线组成, 其余为字母, 数字, 下划线, 且命名不能和系统标识符相同**
- 变量的命名区分大小写
- 不允许使用 C 关键字为变量命名
- 变量名的长度是有限的
- 变量名应简单、容易记忆,应尽量做到见名知意。
- 在同一程序的同一函数中,不能给不同的变量取相同的变量名。

6. 字符常量那个和字符串常量的区别是什么？

- 字符常量:用单撇号括起来的一个字符,如, 'A', 'a'等
- 字符串常量:用一对双撇号括起来的字符序列,如"boy","123"等

7. i++和++i 的区别？

- i++:先使用 i 的值, 再执行 i=i+1 操作。
- ++i:先执行 i=i+1 操作,再使用 i 的值。

8. C 语言的基本数据类型有几种？ 分别占几个字节？

- C 语言的基本数据类型主要有:char, int, float, double 等。char 为字符型,占 1 个字节;int 为整型,占 4 个字节;float 为单精度浮点型,占 4 个字节;double 为双精度浮点型占 8 个字节

9. 请说明算法的五个基本特征, 并进行简要的分析。

- (1)输入:在算法中可以有零个或者多个输入。
- (2)输出:在算法中至少有一个或者多个输出。
- (3)有穷性:任意一个算法在执行有穷个计算步后必须终止。
- (4)确定性:算法的每一个步骤都且有确定的含义,不会出现二义性。
- (5)可行性: 算法的每一步都必须是可行的。

10. 算法的标识方法有哪些？

- a. 自然语言
- b. 流程图
- c. N-S 图
- d. 伪代码
- e. 计算机语言

11. 简述算法的三种基本控制结构。

- a. 顺序结构:按照语句的先后顺序执行即为顺序结构,是算法中最简单的一种结构
- b. 选择结构:根据条件的不同,程序执行不同的语句, 即为选择结构
- c. 循环结构:根据约束条件的不同,多次重复执行某一条或多条语句的结构,称为循环结构

12. if 选择结构与 switch 选择结构的区别是什么？

- if 选择结构一般用于单分支选择结构根据判断的结果决定执行哪种操作
- switch 语句一般用于实现多分支选择结构,可以根据表达式的值,使流程跳转到不同的语句。

13. C 语言中如何表示“真”和“假”系统如何判断？

- C 语言中用“1”表示“真”,用“0”表示“假”。系统在判断时,一般是当表达式的值非“0”时,即表示“真”;仅当表达式的值为“0”时,表示“假”

14. 当(while)型循环结构和直到(until)型循环结构的主要区别是什么？

- 当(while)型循环结构是先判断条件然后再运行,如果条件不成立,则循环可能一次也不执行;
- 直到(until)型循环结构是先无条件执行一次循环体,然后再检查条件,若成立则继续执行循环体也就是说,直到型(until)循环结构无论条件是否成立, 都至少执行一次循环体。

15. break 和 continue 的区别是什么？

break 提前退出循环, continue 结束本次循环

16. 要构成有效循环,则必须满足的条件是什么?

要构成有效的循环,则必须满足以下两个条件:

- (1)循环体,必须含有需要重复执行的操作;
- (2)循环结束的条件,即在什么条件下停止重复。

17. 什么是字符数组, 输入输出方法有哪些?

- 在 C 语言中,没有专门的字符串类型,通常用一个用来存放字符的数组来存放字符串称为字符数组。
- 对字符数组进行输入可以调用 scanf()函数和 gets()函数,输出字符数组可以调用 printf()函数和 puts()函数。

18. 简述定义一维数组的一般形式,并试着定义一个具有 0 个元素的数组 a,用{1,2,3}"对其进行初始化。

- 定义一维数组的一般形式为: 类型符 数组名[常量表达式]
- 定义和初始化数组: int a[10] = {1,2,3}

19. 简述输入, 输出字符串的方法。

输入字符串: gets 方法

输出字符串: puts 方法

20. 对于一个字符数组简述赋值语句和 strcpy 函数的区别

- 不能直接使用赋值语句将一个字符串常量或字符数组复制到一个字符数组中。赋值语句只能将一个字符赋给一个字符型变量或字符数组元素,若要将一个字符串复制到另一个字符数组中, 必须使用 strcpy 函数。

21. real, 什么是局部变量? 什么是全局变量?

22. 简述分别从用户使用的角度和函数的形式可以将函数分为哪几类?

- **从用户使用的角度看可将函数分为库函数和用户自定义函数两种;从函数的形式看可以将函数分为无参函数和有参函数两类。**

23. 简述定义无参函数和有参函数的一般形式、

- 无参函数的一般形式为: 类型名 函数名() { 函数体 }
- 有参函数的一般形式为: 类型名 函数名(形式参数表列){ 函数体 }

24. 简述函数调用的 3 种方法。

- 1.函数语句 2.函数表达式 3.函数参数

25. 简述实参和形参的数据传递过程

- 实参向形参的数据传递是“值传递”, 单向传递, 只能由实参传给形参, 不能由形参传给实参, 实参和形参在内存中占有不同的存储单元, 实参无法得到形参的值

26. 简述变量的存储方式

- 变量由静态存储和动态存储两种方式, (1)静态存储方式是指在程序运行期间由系统分配固定的存储空间的方式。(2)动态存储方式是指在程序运行期间根据需要进行动态分配存储空间的方式

27. real, 什么是指针? 什么是指针变量? 他们有什么关系?

- 变量名与变量的地址有一一对应的关系,可以按此地址直接对变量的存储单元进行访问,这个指向变量的地址叫作指针。专门用来存放另一变量的地址(即指针)的变量称为指针变量。

28. 指针变量与整型常量的加减运算代表什么意义?

- 指针变量与整型常量的加减表示移动指针,以指向当前目标前面或后面的若干个的目标。指针与整型常量 i 的加减等于指针值(地址)与 $i * \text{sizeof}(\text{目标类型})$ 积的加减,得出新的地址。

29. 设 a 为数组名,那么 $a++$ 是否合法?为什么?

- 若 a 为数组名,则 $a++$ 非法,因为 a 是指针常量,不能进行自增。

30. 简述指针、指针变量与地址的关系

- 指针是一个地址,指针变量是用来存放地址的变量,指针变量的值是地址(即指针)

31. 简述使用字符指针变量和字符数组的区别。

- (1) 字符数组由若干个元素组成,每个元素中放一个字符,而字符指针变量中存放地址(字符串第 1 个字符的地址)。
- (2) 可以对字符指针变量赋值,但不能对数组名赋值。
- (3) 编译时为字符数组分配若干存储单元,以存放各元素的值,而对字符指针变量,只分配一个单元。
- (4) 指针变量的值是可以改变的,而数组名代表一个固定的值(数组首元素的地址),不能改变

32. 什么是结构体?结构体类型是如何定义的?

- C 语言允许用户自己建立由不同类型数据组成的结合型的数据结构,称为结构体。
声明一个结构体类型的一般形式为:

```
/*
struct 结构体名
{ 成员表列 };
*/
```

33. 与数值型数组相比,结构体数组具有什么不同之处?

- 每个数组元素都是一个结构体类型的数据, 它们都分别包括各个成员项

34. 简述共用体类型的概念及其特点。

- 使几个不同不同变量共享同一段内存的结构, 称为“共用体”类型的结构
- 特点: 共用体变量所占的内存长度等于最长的成员的长度, 不能引用共用体变量, 而只能引用共用体变量中的成员。

35. 什么是文件型指针?

- 缓冲文件系统中,关键的概念是文件类型指针,简称“文件指针”。每个被使用的文件都在内存中开辟一个相应的文件信息区,用来存放文件的有关信息(如文件的名字、文件状态及文件当前位置等)。这些信息是保存在一个结构体变量中的。该结构体类型是由系统声明的,命名为 FILE

36. 通过文件指针访问文件有什么好处?

- 通过文件指针访问文件的好处是;可以随机访问文件,有效表示数据结构,动态分配内存,方便使用字符串,有效使用数组。

37. 对文件的打开与关闭的含义是什么?

- “打开”是指为文件建立相应的信息区(用来存放有关文件的信息)和文件缓冲区(用来暂时存放输入输出的数据)。

38. 什么是文件的顺序读写?

- 对顺序读写来说,对文件数据的读写顺序和数据在文件中的物理顺序是一致的。在顺序写时,先写人的数据存放在文件中前面的位置,后写人的数据存放在文件中后面的位置;在顺序读时,先读文件中前面的数据,后读文件中后面的数据。

39. 什么是文件的随机读写?

- 随机访问不是按数据在文件中的物理位置次序进行读写,而是可以对任何位置上的数据进行访问。

40. 什么是文件位置标记?它的作用是什么?

- 为了对读写进行控制,系统为每个文件设置了一个文件读写位置标记(简称文件位置标记或文件标记),用来指示“当前的读写位置”。

41. real 什么是数据, 什么是数据对象?

- 数据是客观事物的符号表示, 是所有能输入到计算机中并被计算机程序处理的符号的总称。
- 数据对象是性质相同的数据元素的集合,是数据的一个子集。

42. real 什么是数据结构? 什么是数据类型

- (1)数据结构。数据结构是相互之间存在一种或多种特定关系的数据元素的集合, 也可以说数据结构是带“结构”的数据元素的集合, “结构”是指数据元素之间存在的关系。
- (2)数据类型。数据类型是一个值的集合和定义在这个值集上的一组操作的总称

43. 什么是数据的存储结构? 数据的存储结构又分为哪两种?

- 数据对象在计算机中的存储表示称为数据的存储结构,也称为物理结构。
- 数据元素在计算机中有两种基本的存储结构,分别是顺序存储结构和链式存储结构。

44. 数据元素与数据项的区别是什么?

- 数据元素是数据的基本单位,有时也称为元素、记录等。在计算机中通常作为一个整体进行考虑和处理,数据元素用于完整地描述一个对象。数据项是组成数据元素的、有独立含义的、不可分割的最小单位。

45. 什么是抽象数据类型?它包含哪几个部分?

- 抽象数据类型一般指由用户定义的、表示应用问题的数学模型, 以及定义在这个模型上的一组操作的总称, 具体包括 3 个部分:数据对象、数据对象上关系的集合以及对数据对象的基本操作的集合。

46. 非空线性表或线性结构具有什么特点?

(1)存在唯一的一个被称作“第一个”的数据元素;(2)存在唯一的一个被称作“最后一个”的数据元素;(3)除第一个数据元素之外,结构中的每个数据元素均只有一个前驱;(4)除最后一个数据元素之外,结构中的每个数据元素均只有一个后继

47. 什么是循环链表?

在用指针实现表时,表中最后一个结点的指针为空指针,如果将最后一个结点的指针域指向头结点,使整个链表形成一个环,这种首尾相接的链表就称为循环链表。在循环链表中, 从任意一个结点出发都可以找到表中其他结点。

48. 怎样区分头指针, 头结点, 首元节点

- 首元结点是指链表中存储第一个数据元素的结点

- 头结点是在首元结点之前附设的一个结点,其指针域指向首元结点。头结点的数据域可以存储与数据元素类型相同的其他附加信息,也可不存储任何信息。
- 头指针是指向链表中第一个结点的指针。若链表设有头结点,则头指针所指结点为线性表的头结点;若链表不设头结点,则头指针所指结点为首元结点。

49. 请简要比较顺序表和链表。

- 存储分配的方式:顺序表的存储空间是静态分配的;链表的存储空间是动态分配的
- 存储密度(存储密度=结点值域所占的存储量/结点结构所占的存储总量)顺序表的存储密度=1;链表的存储密度<1(因为结点中有指针域)。
- 存取方式:顺序表可以随机存取,也可以顺序存取;链表是顺序存取的。
- 插入或删除操作时移动元素的个数:顺序表平均需要移动近一半元素;链表不需要移动元素,只需要修改指针。

50. 什么是栈?栈的修改原则是什么?

- 栈是限定仅在表尾进行插入或删除操作的线性表。栈的修改是按后进先出的原则进行的。

51. 什么是链栈?链栈相比于顺序栈来说最大的优点是什么?

- 链栈是指采用链式存储结构实现的栈。链栈的容量理论上是无限的,不像顺序栈受到

52. 采用“分治法”求解递归问题需要满足的条件是什么?

- a. 能将一个问题转变成一个新问题,而新问题与原问题的解法相同或类同,不同的仅是处理对象,并且这些处理对象更小且变化有规律。
- b. 可以通过上述转化而使问题简化
- c. 必须有一个明确的递归出口,或称递归的边界。

53. 什么是队列?队列的修改原则是什么?

- 队列是一种特殊的线性表,只在表头(也称为队头)进行删除操作,只在表尾(也称为队尾)进行插入操作。队列的修改是按先进先出的规则进行的。

54. 循环队列的优点是什么?如何判别它的空和满?

- 循环队列可以克服顺序队列的“假上溢”现象,能够使存储队列的向量空间得到充分的利用。

55. real 什么是串?什么是子串?

56. 简述子串的定位运算。

- 子串定位运算类似于串的基本运算中的字符定位运算。只不过是找子串而不是找字符在主串中首次出现的位置。子串定位运算又称串的模式匹配或串匹配。

57. 什么是矩阵的压缩存储?

- 在数值分析中经常出现一些高阶矩阵,在这些高阶矩阵中有许多值相同的元素或者是零元素,为了节省存储空间,对这类矩阵采用多个值相同的元素只分配一个存储空间、零元素不存储的存储策略称为矩阵的压缩存储。

58. 什么是广义表?什么是广义表的表头和表尾?

- 广义表是 $n(n>0)$ 个元素 $a_1, a_2, \dots, a_i, \dots, a_n$ 组成的有限序列。若广义表 LS 非空($n>1$),则 a_1 是 LS 的表头,其余元素组成的表($a_2, \dots, a_i, \dots, a_n$)称为 LS 的表尾。

59. 简述广义表的深度和长度。

- 广义表的深度是指表展开后所含括号的层数。

- 广义表的长度是指广义表中所包含的数据元素的个数。
60. real, 什么是树的路径长度? 什么是树的带权路径长度?
61. real, 什么是二叉树?请简述二叉树的五种基本形态.
62. 什么是叶子结点?什么是分支结点?
- 度为 0 的结点称为叶子结点, 也称为终端结点
 - 度不为 0 的结点称为分支结点, 也称为非终端结点
63. 什么是结点的层数?什么是树的高度?
- 规定根的层数为 1, 对其余任何结点, 若某结点在第 k 层, 则其孩子结点在第 $k+1$ 层
 - 树中所有结点的最大层数称为树的高度
64. 简述二叉树的特点
- a. 每个结点最多有两棵子树,所以二叉树中不存在度大于 2 的结点; 2.子树的次序不能任意颠倒,某结点即使只有一棵子树也要区分是左子树还是右子树。
65. 简述满二叉树的概念及其特点
- 概念: 在一棵二叉树中,如果所有分支结点都在左子树和右子树, 并且所有叶子结点都同一层上, 这样的二叉树称为满二叉树。
 - 特点: 1.叶子结点都在最下面一层上;2.只有度为 0 和度为 2 的结点。
66. 简述完全二叉树的概念及其特点。
- 概念: 对于一颗具有 n 个结点的二叉树按层序编号, 如果编号为 $i(1 \leq i \leq n)$ 的结点与同样深度的满二叉树中的编号为 i 的结点在二叉树中的位置完全相同, 则这颗二叉树称为完全二叉树
 - 特点: 1. 叶子结点只能出现在最下面两层上, 且最下层叶子结点都集中在左边连续的位置。2. 如果有度为 1 的结点, 只可能有一个, 且该结点只有左孩子。
67. 简述拓扑排序的过程
- (1)在有向图中选一个无前驱的顶点且输出它。
 - (2)从图中删除该顶点和所有以它为尾的弧。
 - (3)重复(1)和(2),直至不存在无前驱的顶点
 - (4)若此时输出的顶点数小于有向图中的顶点数,则说明有向图中存在环,否则输出的顶点序列即为一个拓扑序列。
68. 简述有向完全图和无向完全图的概念。
- 对于含有 n 个顶点的无向图,若具有 $n(n-1)/2$ 条边,则称为无向完全图。
 - 对于含有 n 个顶点的有向图,若具有 $n(n-1)$ 条弧,则称为有向完全图。
69. 简述邻接矩阵表示法的优缺点
- 优点:1.便于判断两个顶点之间是否有边。2.便于计算各个顶点的度
 - 缺点: 1.不便于增加和删除顶点。2.不便于统计边的数目,需要扫描邻接矩阵所有元素才能统计完毕, 时间复杂度为 $O(n^2)$ 。3 空间复杂度高
70. 简述邻接表表示法的优缺点
- 优点: 1.便于增加和删除顶点。2.便于统计边的数目,按顶点表顺序扫描所有边表可得到边的数目。3.空间效率高。
 - 缺点:1.不便于判断顶点之间是否有边。2.不便于计算有向图各个顶点的度

71. 简述顺序查找的基本思想以及优缺点。

- 基本思想：顺序查找的基本思想是:从表的一端开始,顺序扫描线性表,依将扫描到结点的关键字和给定值 K 进行比较,若当前扫描到结点的关键字与 K 相等,则查找成功;若扫描结束后,仍未找到关键字等于 K 的结点,则查找失败。
- 优点：**算法简单，对表结构无任何要求，既适用于顺序结构，也适用于链式结构，无论结点之间是否按关键字有序排列，它都适用。**
- 缺点：查找效率低，当数据较多时不宜采用顺序查找。

72. 简述折半查找的优点和缺点。

- 优点:比较次数少,查找效率高。
- 缺点:对表结构要求高，只能用顺序存储的有序表。

73. 简述分块查找的优点和缺点。

- 优点：在表中插入和删除数据元素时，只要找到该元素对应的块,就可以在块内进行插入和删除运算。由于块内是无序的,所以插入和删除比较容易,无需大量移动数据元素。如果线性表既要快速查找又经常动态变化,则可采用分块查找。
- 缺点：要增加一个索引表的存储空间并对初始索引表进行排序运算。

74. 构造散列函数的方法有哪些？

- 构造散列函数的方法有：**除留余数法、数字分析法、平方取中法和折叠法。**

75. 简述线性探测法、二次探测法和伪随机探测法的优缺点

- 线性探测法优点：只要散列表未填满，总能找到不友生冲突的地址;
- 线性探测法缺点：会产生“二次聚集”现象;
- 二次探测法和伪随机探测法的优点是：可以减少“二次聚集”现象;
- 二次探测法和伪随机探测法的缺点是：不能保证一定找到不发生冲突的地址。

76. 什么是稳定的排序方法?什么是不稳定的排序方法?

- 针对排序记录中的两个相等关键字,它们在序列中的相对位置，在排序之前和经过序之后,没有改变,则该排序方法就是稳定的;若它们的相对位置发生改变,则该排序方法就是不稳定的

77. 常见的稳定的排序方法有哪几种?不稳定的排序方法有哪儿种?

- 稳定排序方法：直接插入排序、冒泡排序、简单选择排序、归并排序;
- 不稳定排序方法：希尔排序、快速排序、堆排序

78. 什么是内部排序?什么是外部排序?

- 内部排序是指待排序记录全部存放在计算机内存中进行排序,不需要访问外存,适于排序记录较少的情况;
- 外部排序是指待排序记录的数量很大,内存无法容纳全部数据,在排序的过程中要借助外存的排序。

79. 什么是交换类排序?可举例说明。

- 交换类排序是通过“交换”无序序列中的记录从而得到其中关键字最小或最大的记录并将它加入有序子序列中,以此类方法来增加记录的有序子序列的长度。主要包括冒泡排序和快速排序。

80. 简述冒泡排序第一趟排序的过程

- 冒泡排序第一排序的过程是:待排序的记录存放在数组 $r[1...n]$ 中。首先将第 1 个记录的关键字和第 2 个记录的关键字进行比较,若为逆序,则交换两个记录。然后比较第 2 个记录和第 3 个记录的关键字。依次类推,直至第 $n-1$ 个记录和第 n 个记录的关键字进行过比较为止, 这样便完成了 1 趟排序

81. 在选择合适的排序方法时,应考虑的因素有哪些?

- (1)待排序的记录个数;
- (2)记录本身的大小;
- (3)关键字的结构及初始状态;
- (4)对排序稳定性的要求;
- (5)存储结构。