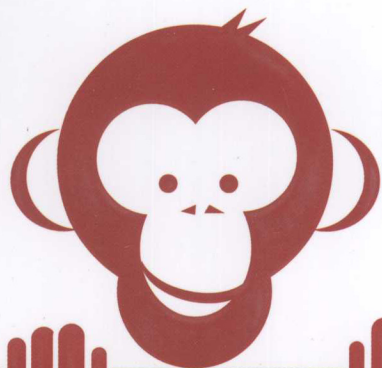


开点工作室 / 编著

横扫



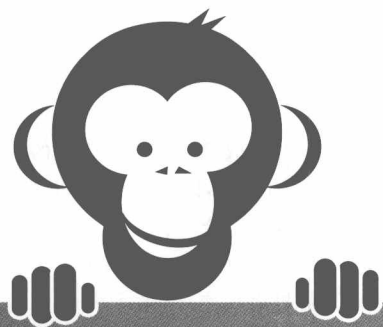
offer

程序员招聘真题详解700题



清华大学出版社





横扫Offer

程序员招聘 真题详解700题

开点工作室 编著

清华大学出版社
北京

内 容 简 介

本书对近年来 IT 企业技术类招聘笔试、面试的试题进行全面的收集和梳理,并选取其中具有代表性的 700 道真题进行深入的解析,力图将其中所蕴含的计算机专业知识进行系统的归纳总结,既可以作为高等学校计算机类专业学生复习和练习的参考资料,也可以作为求职者快速学习和备考的综合教程。

本书内容按照目前各主流 IT 企业技术类招聘考试试题所涉及的专业知识领域,分为 8 章:第 1 章 C 语言程序设计,第 2 章 面向对象程序设计,第 3 章 数据结构与算法,第 4 章 计算机组成原理,第 5 章 操作系统,第 6 章 计算机网络,第 7 章 数据库技术,第 8 章 数学与逻辑。各章直接对招聘考试真题展开解析,将试题涉及的知识点进行精准的归纳和深入的阐述,而省去了与之不相关的教材内容的堆砌,确保应聘者的学习更具有针对性。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

横扫 Offer: 程序员招聘真题详解 700 题/开点工作室编著. --北京:清华大学出版社,2016

ISBN 978-7-302-44568-5

I. ①横… II. ①开… III. ①程序设计—工程技术人员—招聘—考试—题解 IV. ①TP311.1-44

中国版本图书馆 CIP 数据核字(2016)第 175225 号

责任编辑:张瑞庆

封面设计:常雪影

责任校对:李建庄

责任印制:沈 露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 刷 者:北京鑫丰华彩印有限公司

装 订 者:三河市吉祥印务有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:30.75

字 数:745 千字

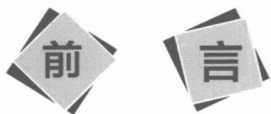
版 次:2016 年 9 月第 1 版

印 次:2016 年 9 月第 1 次印刷

印 数:1~2000

定 价:59.50 元

产品编号:070299-01



在 2015 年 3 月召开的全国两会上,李克强总理在政府工作报告中首次提出了“互联网+”的概念,标志着以互联网为代表的信息技术与传统行业全面融合的新型经济发展模式已经上升到了国家经济的战略层高度。金融、商业、教育、交通、医疗等各种传统行业,一旦插上了互联网的翅膀,便会展现出惊人的爆发力,创造出一个又一个科技和财富的神话。当阿里巴巴、腾讯、百度、小米等企业已经成为市值千亿的超级公司时,当每个月都有各种互联网创业企业获得千万级风险投资时,当人们每天的衣食住行已经离不开微信、淘宝、支付宝、滴滴打车、大众点评这些 APP 时,我们知道,一个新的时代开启了。

“互联网+”风潮的到来,不仅是聚集财富资本和催生新型商业模式的过程,而且还提供了大量的 IT 专业技术工作岗位。各种概念、模式和奇思妙想最终都要通过一行行的计算机程序代码,变成形形色色的软件、网站、APP,才能改变人们的生活,为企业创造物质财富。对于广大计算机类专业的大学毕业生和想投身 IT 产业的年轻人,现在无疑正面临着更多、更好的机会。IT 类专业求职者眼中的理想公司已经不再局限为 IBM、微软、甲骨文等国外老牌软硬件企业,以 BAT 为代表的国内互联网巨头,正在吸引着越来越多的优秀 IT 专业人才。同时,各种创新创业型中小企业,凭借丰厚的薪酬待遇和股权激励以及广阔的发展空间,也在积极地招兵买马,准备大干一场。在 2015 年这个“史上最难就业季”中,计算机专业仍然是最容易找工作、平均薪酬最高的专业之一。

互联网产业的繁荣和 IT 就业市场的火爆,并不意味着好工作就会自动落到每一个求职者的头上,企业在提供高薪岗位的同时,也对求职者的专业素质和能力提出了很高的要求。特别是像 BAT、小米、华为、奇虎 360 等名企的招聘过程,往往要进行多轮的笔试和面试,应聘者要经过激烈的竞争才能最终获得满意的 offer。

从一名计算机专业的学生成为一名合格的程序员,并不是一件很容易的事情,你需要接受系统的计算机专业教育,通常需要学习几十门专业课程,进行大量的软硬件实验,最好有机会参与实际项目的开发,这些努力和付出会让你掌握大量的专业知识,具备一定的专业技能和经验。最后,你还要把专业知识、技能、经验和素养尽可能充分地展示和证明给企业,从而让企业能够知道你可以成为一名合格的程序员。形形色色的 IT 企业笔试和面试便是检验和测量应聘者专业知识和能力的最直接的工具,也是大多数应聘者获得 offer 必须要跨过的一道门槛。要想获得心仪 IT 企业的工作机会,除了优异的专业课成绩、真实可信的项目经历外,认真对待和积极准备各大名企的笔试、面试也是应聘者取得成功的关键因素。

和其他各种考试项目一样,IT 企业的技术类招聘考试也是一种心理测量形式,其考试本身应该具备一些固定的属性,包括:被测对象、考试内容、考查目标、考试难度等。有了这些属性,就可以比较全面的描述某一项考试。通过对近几年各大知名 IT 企业技术类招聘笔试、面试试题的分析,我们总结出这类考试具有以下两个特点:

(1) 考试内容相对发散,没有明确的考试大纲、考试说明和教程可以参照。招聘企业的类型、产品、岗位各不相同,因此其笔试、面试的考试内容不局限于某一个知识领域,而是涉及各种计算机软硬件理论、方法和技术。通常情况下,企业也不会专门为自己的招聘考试编写考试大纲或考试说明,指定专门的考试指导用书。应聘者需要专门拿出时间和精力对计算机学科的多门专业基础课程进行复习和准备。

(2) 试题着重考查应聘者对专业基础知识的领会、综合、运用和迁移的情况,体现计算机学科逻辑思维和计算思维的能力目标。企业招聘考试中大部分试题所涉及的知识点都是计算机专业各学科最核心和最基础的内容,但应聘者通过“死记硬背”的方式往往是解答不出这些试题的,需要对这些基础知识真正的理解,并且能够在各种实际场景中综合应用这些知识来解决具体问题。

开点工作室是由一群重点高校计算机专业中青年骨干教师组成的教育教学兴趣小组。热爱计算机教学,喜欢钻研专业知识,关注学生的成长发展,是我们共同的特点。每年,我们都会目睹自己身边的学生求职找工作的过程,通过和他们的交流,我们深深感受到 IT 企业,特别是知名企业的招聘考试考查范围宽泛,试题形式灵活,综合性强,对于应聘者的要求是比较高的。学生在准备招聘考试的过程中,往往会遇到需要复习的内容纷繁众多,缺少针对性参考资料,没有真题的解析和练习等困惑。尤其是对于非计算机专业科班出身的学生,要让他们在短时间内熟练掌握多门计算机专业课程的知识是很困难的。为此,我们非常希望利用自身的专业知识和教学经验,编写出这样一部书籍:通过对近年来 IT 企业技术类招聘笔试、面试的试题进行全面的收集和梳理,并选取其中具有代表性的试题进行深入的解析,力图将其中所蕴含的专业知识进行系统的归纳总结,既可以作为计算机专业学生复习和练习的参考资料,也可以作为非计算机专业求职者快速学习、备考的综合教程。

本书精心选取了 700 道各 IT 名企程序员招聘笔试面试的真题,书中内容按照目前各主流 IT 企业技术类招聘考试试题所涉及的专业知识领域划分为 8 章:第 1 章为 C 语言程序设计,第 2 章为面向对象程序设计,第 3 章为数据结构与算法,第 4 章为计算机组成原理,第 5 章为操作系统,第 6 章为计算机网络,第 7 章为数据库技术,第 8 章为数学与逻辑。各章直接对招聘考试真题展开解析,将试题涉及的知识点进行精准地归纳和深入地阐述,而省去了与之不相关教材内容的堆砌,确保应聘者的学习更加有针对性。我们希望本书的读者能够把书中每一道试题的解析读懂、吃透,从而牢固掌握试题所涉及相关知识的内涵、外延和综合应用,真正做到举一反三。同时,本书专注于计算机专业技术试题的分析讲解,不涉及性格测试、智力测验、常识测试、英语测试以及面试礼仪等非技术领域内容,真正做到“没有鸡汤,全是干货”。书中招聘考试真题来自互联网中求职网站、论坛、技术博客等渠道,还包括参加招聘考试学生的内容回忆,并且加入了其他领域有参考价值的试题,如研究生统考(408)中的部分试题。有些原始试题由于各种原因,内容不够完整或叙述不够准确,作者进行了适当的补充和修正,以免造成对读者学习的迷惑和误导。

“拨开学习中的荆棘迷雾,指点探索中的灵感思路”,希望这本凝结了集体智慧和心血的教材,能够充分体现开点工作室的创作主旨,为广大 IT 求职者提供真正的帮助。希望每一名热爱这个行业,充满热情和力量的年轻人都能梦想成真。

由于水平有限,加之时间仓促,书中肯定会有这样或那样的错误和不足,敬请读者提出宝贵意见,以便帮助我们修改完善。作者电子信箱: kaidiancs@163.com。

更多计算机知识学习、技术讲解、职业规划和求职指导的文章请扫描以下二维码,关注我们的微信公众号:开点工作室。



长按二维码,可以关注我们哟!

微信ID: kaidiancs

欢迎您关注开点工作室!这

里有大量的技术专题文章、

IT企业招聘真题解析和专业

书籍点评推荐,很高兴能和

您分享计算机专业知识教与

学过程中的经验和体会。

开点工作室

2016年5月于北京

第 1 章

C 语言程序设计



导 读

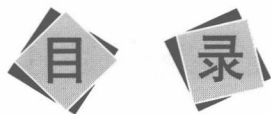
“C 语言程序设计”几乎是全国理工科高校学生的计算机入门语言,也是近几年中许多 IT 企业在选拔人才过程中必考的内容之一。企业选拔人才进行笔试和面试过程中,考核的重点与难点与一般的考试不同,他们更注重各项知识在实际工程环境中的运用,更注重应聘者对基本概念和算法的深入理解和分析。

企业笔试和面试过程中涉及 C 语言相关的内容很多,要求的知识面也很宽,绝不局限于大学中一门程序设计课程的教学内容,更多的是结合企业工作中的实际应用来考察“企业认为”的知识点。分析部分企业的笔试和面试内容可以看出,企业关心的知识点与学校教学的知识点并不完全重合,企业考核的重点也不是学校考核的重点,甚至于企业考核的知识点在学校的教学中都不会涉及。下面列出的是企业在进行人才选拔时与 C 语言相关的考核重点和难点。

(1) 基本概念。除常规的一些基本内容之外,企业更会考核一些具有 C 语言特色的编程方式。例如,不同的数据类型在内存中的编码形式与表示范围,++、-- 运算的前缀/后缀形式,关系运算与(&&)、或(||)在运算过程中的计算“短路”现象, sizeof 运算等。这些考核点均是 C 语言的基本概念,也是容易被忽略、容易出错的知识。

(2) 编译预处理。各种预处理命令,宏定义与宏替换等。这是学校教学中几乎没有涉及或很少涉及的内容,而各种预处理指令是在编写大型软件系统中必须使用的编程命令,合理使用预处理命令也是编程技巧。

(3) 指针综合运用。通过指针访问一般变量,通过指针访问指针变量,通过指针操作字符串,通过指针访问二维数组,通过指针访问函数、结构等,指针数组和数组指针,函数返回指针等。指针是 C 语言中最活跃的成分,它的千变万化既体现了指针的特性,也可以充分反映被测者的基本功,企业在考核学生基本功时,都会选择指针及相关内容作为考核点。



第1章 C语言程序设计

VII

导读	VII
1.1 基本概念与基本运算	2
1.2 基本语句	10
1.3 函数	16
1.4 数组	23
1.5 指针	31
1.6 结构与联合	50
1.7 位运算	58
1.8 编程题	64

第2章 面向对象程序设计

82

导读	82
2.1 基本概念	83
2.2 类和对象	97
2.3 继承、派生和多态	104
2.4 模板	114

第3章 数据结构与算法设计

118

导读	118
3.1 线性表	120
3.2 栈、队列和数组	131
3.3 树与二叉树	158
3.4 图	181
3.5 查找	192
3.6 排序	208
3.7 时间复杂度	225
3.8 数字排列	230

第4章 计算机组成原理

234

导读	234
4.1 计算机系统概述	236
4.2 数据的表示与运算	237
4.3 指令系统与处理器组织	252
4.4 存储器组织	255
4.5 输入输出组织	264
4.6 计算机系统综合	271

第5章 操作系统

285

导读	285
5.1 进程管理	286
5.2 内存管理	313
5.3 文件管理	322
5.4 设备管理	327

第6章 计算机网络

334

导读	334
6.1 应用层	336
6.2 传输层	346
6.3 网络层	357
6.4 数据链路层与局域网	375
6.5 网络与通信基础	386
6.6 网络安全	393

第7章 数据库技术

398

导读	398
7.1 基本概念	400
7.2 SQL 语言	408
7.3 数据库设计	417
7.4 数据库管理	418
7.5 查询优化与索引	422
7.6 综合应用	425
7.7 其他类型题目	434

第 8 章 数学与逻辑

437

导读	437
8.1 排列组合	438
8.2 概率统计	447
8.3 逻辑推理	463

参考文献

480

(4) 位运算。基本概念与相关运算(位与、位或、位非、位异或、位左移、右移等),使用位运算和相关知识设计算法。例如,将二进制中指定位置 1,通过二进制操作实现各种算术运算等。位运算所具有的独特性质,会带来许多有意思的问题和解决方案,所以,不少面向硬件与系统级开发的企业,都会选择这方面的内容进行考核。

(5) 常见典型的基本算法。典型的字符串操作(求串长、复制、连接、比较、求子串等),交换变量值等问题。本类题目的程序不长,但对经典问题的掌握程度可以较好地反映出应聘者的基础。

(6) 简单应用的算法设计。从实际工作中提炼出来的简单应用,例如,按特殊要求排序、手机号码合法性、与日期相关处理、输出指定的简单图形等。这些题目的程序可能都不太长,但关键是要对问题本身有一个透彻的分析,并在分析的基础上设计出精巧的算法。这是考核分析问题进行算法设计的能力的有效手段。

(7) C 语言与其他内容(课程)的融合。例如,不同存储类型的变量占用内存情况(可能涉及计算机组成原理、操作系统或编译),不同数据类型的变量在内存的边界对齐情况(可能涉及计算机硬件结构或编译器),用加法运算等基本操作实现乘法(模拟 CPU 的基本操作),在 Windows 中控制任务管理器的 CPU 占用率(涉及操作系统),同一程序在不同编译环境下运行等。这些题目均可能涉及传统高校的不同课程,多课程知识点的融合更是企业考察被测人员能力的常用的和有效的手段。

与 C 语言程序设计相关的题目非常多,同一问题的变化形式也非常多,打好基础才是应对的关键。学习 C 语言不能停留在纸面学习,一定要上机编程实践。大量的编程实践可以强化对于基本概念的理解和认识,强化对经典算法的理解,强化算法设计,所以在准备笔试和面试阶段,可以编写一些典型的程序作为练手和热身。

1.1 基本概念与基本运算

题 1.1

哪些是将一个十六进制赋值给一个 long 型变量()。

A. long number=0345L

B. long number=345L

C. long number=0345

D. long number=0x345L

【出 处】腾讯。

【答 案】D。

【知识点】整型常数的表示形式。

【解 析】C 语言中规定,常数以数字 0 开头是八进制形式,以 0x 或 0X 开头为十六进制形式,常数的后面加字母 L 表示是长整数。

题 1.2

执行 C 程序代码: int a=1;int b=0;int c=0;int d=(++a)*(c=1);,则 a,b,c,d 的值分别为()。

A. 2,0,1,2

B. 1,0,1,1

C. 2,0,1,1

D. 2,0,0,2

【出 处】浙江大华,2015。

【答 案】A。

【知识点】整型常数的表示形式。

【解 析】表达式 $d=(++a)*(c=1)$ 的执行过程是: 先进行 ++ 运算,变量 a 为 2,再进行 $(c=1)$ 运算,变量 c 为 1,变量 d 结果为 2。

题 1.3

“零值”可以是 0、0.0、FALSE 或者“空指针”。例如,int 变量 n 与“零值”比较的 if 语句为: if (n==0)。BOOL flag 与“零值”比较的 if 语句为: (①);float x 与“零值”比较的 if 语句为: (②)。

【出 处】中兴。

【知识点】数据类型与表示形式。

【答 案】①if (flag) ②if (x==0.0)

【解 析】对于 BOOL 类型的变量 flag,不能使用类似 $flag==TRUE$ 或者 $flag==FALSE$ 这样的条件判断。对于 float 或 double 类型的变量 x,由于 float 和 double 类型均可以精准表示出“零值”,所以可以直接采用语句 if (x==0.0) 判断 float 变量是否为“零值”。

【拓 展】对于浮点类型(float 或 double 类型)的数据有必要进行更进一步讨论。

浮点型数据在计算机内部采用 IEEE 754 浮点格式标准表示,其尾数和阶码部分的位数都是有限的,因此,在将一个十进制数转换为二进制表示时,如果不能以 IEEE 754 给定的尾

数有效位数精确表示的话,就会发生低位有效位数丢失,造成一定的误差。如果直接用比较运算`==`来判断这种具有误差的值时,可能会导致判断结果错误。

在高级语言程序设计中,对于两个浮点类型的变量 x 和 y ,一般不直接使用语句 `if (x==y)` 来判断 x 是否与 y 相等,而是当 $|x-y|$ 足够小(如 10^{-6})时即认为变量 x 和 y 是相等的。相关判断语句如下:

```
const float EPSINON=0.000001;
if ((x-y>=-EPSINON) && (x-y<=EPSINON))
```

题 1.4

判断 A、B、C、D 4 个表达式是否正确,若正确,写出表达式 a 的值()。

```
int a=4;
```

A. $a+=(a++)$;

B. $a+=(++a)$;

C. $(a++)+=a$;

D. $(++a)+=(a++)$;

【出 处】华为。

【答 案】A 正确, $a=9$ 。B 正确, $a=10$ 。C 错误。D(C++ 语言环境)正确, $a=11$ 。

【知识点】表达式基本概念。

【解 析】选项 A 是先执行 $a+=a$, 变量 $a=8$, 然后再计算 $++$, 结果 $a=9$ 。选项 B 是先计算 $++a$, 使 $a=5$; 再计算 $a+=a$, 结果 $a=9$ 。选项 C 错误。选项 D 如果是在 C 语言中, 则错误; 如果是在 C++ 语言中, 则正确, 其计算过程是: 先计算 $++a$, 使 $a=5$, 再计算 $a+=a$, 使 $a=10$, 最后再计算 $a++$, 结果 $a=11$ 。

【拓 展】题中所给选项的表达式书写风格是一种非常糟糕的编程风格, 这样的编程风格很容易造成歧义, 在实际的软件项目编程中应尽量避免。

题 1.5

下面的代码输出是什么? 为什么?

```
void foo(void)
{   unsigned int a=6;
    int b=-20;
    (a+b>6)?puts(">6"):puts("<=6");
}
```

【答 案】 >6 。

【知识点】有符号类型与无符号类型之间的转换规则。

【解 析】在 C 语言中, 当一个表达式中有 `int` 类型和 `unsigned` 类型两种不同的数据类型时, 所有的操作数都要按照 C 语言表达式数据类型混合运算时的类型自动转换规则将数据转换为无符号整数类型。-20 的机器数为 FFFFFFFECH, 若作为无符号整数解释, 则变成了一个非常大的正整数, 所以表达式 $a+b$ 的结果大于 6。

【拓 展】这一点在频繁使用无符号数据类型的嵌入式系统中更为重要。

题 1.6

以下哪些指令属于在预编译阶段执行(),以下哪些指令属于运算符()。

a: malloc b: ++ c: #pragma d: sizeof e: #define

- A. 预编译 c, e 操作符 a, b B. 预编译 c, d 操作符 b, e
C. 预编译 c, e 操作符 b, d D. 预编译 a, d 操作符 b, d

【出 处】搜狐, 2016。

【答 案】C。

【知识点】编译预处理基本概念。

【解 析】在 C 语言中, 正式开始编译之前要对源程序进行编译预处理, C 语言的编译预处理指令均是有 # 开头的, 选项中 #pragma 和 #define 是编译预处理指令, 在预编译阶段执行。C 语言的操作符是 C 语言语法规则中规定的运算符, 选项 ++ 和 sizeof 是运算符, malloc 是 C 语言标准库中的函数名。

【拓 展】#define 是编译预处理中用于定义符号常量的宏命令, 可以定义符号常量和带有参数的宏定义。

题 1.7

以下程序的输出是多少? ()

```
#define A(x) x+x
int i=5*A(4)*A(6);
cout<<i;
```

- A. 50 B. 100 C. 120 D. 480

【出 处】搜狐, 2016。

【答 案】A。

【知识点】宏替换基本规则。

【解 析】根据 C 语言预处理的宏替换规则, 宏替换的结果是表达式 $i = 5 * x + x * x + x$, 代入参数值后为 $i = 5 * 4 + 4 * 6 + 6$, 计算结果是 50。

题 1.8

以下程序的输出结果是()。

```
#include<stdio.h>
#define add(a,b) a+b
int main()
{ printf("%d\n", 3*add(4,7));
  return 0;
}
```

- A. 33 B. 19 C. 25 D. 49

【出 处】腾讯,2014。

【答 案】B。

【知识点】宏替换基本规则。

【解 析】根据 C 语言的宏替换规则,替换后的表达式是 $3 * a + b$,代入参数值后为 $3 * 4 + 7$,计算结果是 19。



题 1.9

请完成宏定义: `#define swap(a,b) ( ) /* 交换 ab */`。

【出 处】中兴。

【答 案】`#define swap(a,b) {a=a^b;b=a^b;a=a^b;}`。

【知识点】宏替换。

【解 析】交换两个变量的方法有多种,可以采用中间变量,也可以采用位运算,或者是直接加减运算等。这里因为是采用宏定义方式,无法引入合适的中间变量,所以只能使用其他方式。

如果采用直接加减运算,则有: `#define swap(a,b) {a=a+b;b=a-b;a=a-b;}`。

如果采用异或运算,则有: `#define swap(a,b) {a=a^b;b=a^b;a=a^b;}`。



题 1.10

`#include <filename.h>` 和 `#include "filename.h"` 有什么区别?

【出 处】中兴。

【知识点】宏指令。

【参考答案】使用宏指令 `#include <filename.h>` 是通知编译器在预处理时要包含开发环境中提供的名为 `filename.h` 的头文件。使用宏指令 `#include "filename.h"` 是要求编译器在预处理时优先包含编程者指定路径下的名为 `filename.h` 的头文件,如果在指定路径下没有发现名为 `filename.h` 的头文件,则使用系统提供的同名头文件。



题 1.11

头文件中的 `ifndef/define/endif` 有什么用途?

【出 处】中兴。

【知识点】宏指令。

【参考答案】防止该头文件被重复引用。

【解 析】C 语言头文件中的命令 `ifndef/define/endif` 是给编译预处理器的指令,其中 `#ifndef` 是条件编译,用于测试指定的宏是否还没有定义,如果指定的宏还没有定义,则执行直到 `#endif` 为止的宏定义 `#define` 语句序列。`#define` 是用来定义宏。`#endif` 要与 `#ifndef` 配对使用,表示一个条件编译块。

题 1.12

程序运行后的输出结果是()。

```
#include<stdio.h>
int main()
{   int m=0123,n=123;
    printf("%o  %o\n",m,n);
    return 0;
}
```

- A. 0123 0173 B. 0123 173 C. 123 173 D. 173 173

【出 处】蘑菇街,2016。

【答 案】C。

【知识点】整型常数的八进制表示和转换。

【解 析】m=0123 是将采用八进制表示的整型常量赋值给变量 m,n=123 是用十进制表示的整型常量为变量 n 赋值。函数 printf 中的输出格式控制符 %o 的含义是将变量的值按八进制形式输出。

题 1.13

以下代码的执行结果是()。

```
int main()
{   int i=-2147483648;
    return printf("%d,%d,%d,%d\n",~i,-i,1-i,-1-i);
}
```

- A. 0,2147483648,2147483649,2147483647
B. 0,-2147483648,-2147483647,2147483647
C. 2147483647,2147483648,2147483649,2147483647
D. 2147483647,-2147483648,-2147483647,2147483647

【出 处】58 同城。

【答 案】D。

【知识点】整型数据的编码形式与基本运算规则。

【解 析】C 语言中,int 类型占用 32 位二进制位,采用补码形式对整数进行编码,-2147483648 是一个特殊的数值,它是 32 位的 int 类型所能表达的最小值,其二进制补码形式是:10000000,00000000,00000000,00000000。~i 是对变量 i 按位取反,结果为:01111111,11111111,11111111,11111111,是十进制的 2147483647,它是 int 类型能够表示的最大值。-i 是对变量 i 进行求负运算,因为求负=按位取反+1,所以根据上面~i 的结果+1 后,得到的二进制结果是:10000000,00000000,00000000,00000000,即为十进制-2147483648。1-i 的结果是:10000000,00000000,00000000,00000001,即为十进制-2147483647 的补码表示。-1 的二进制补码表示为:11111111,11111111,11111111,11111111。-1-i 的运算过程是:

-1: 11111111,11111111,11111111,11111111

-i: 10000000,00000000,00000000,00000000

结果: 01111111,11111111,11111111,11111111

运算结果即为十进制 2147483647 的补码表示。

【拓展】本题中另一个有意思的知识点在 return 语句,该语句是将函数 printf 的返回值作为函数 main 的返回值进行返回。一般情况下,在编程中并不关心函数 printf 的返回值,故很多人并不知道函数 printf 还会有返回值,会以为这里是程序的一个 bug。

题 1.14

已知: `int i=10,j=10,k=3;k*=i+j;`那么 k 最后的值是多少?

【出处】华为。

【答案】60。

【知识点】运算符 `*` 的基本概念。

【解析】语句 `k*=i+j` 的含义是 `k=k*(i+j)`,则 `k=3*(10+10)=60`。

题 1.15

程序员规范中要求不要写出类似 `(++i)+(i++)` 或 `f(++i,i++)` 这样的代码,请说明原因。

【出处】中兴。

【知识点】编程风格与规范。

【参考答案】表达式的计算顺序是由编译器决定的,虽然参数的压栈顺序在给定的调用方式下是固定的,但参数表达式的计算顺序也是由编译器决定的。不同的编译器对不同的表达式计算的顺序可能不一致。因此,在编程过程中要避免使用与编译器密切相关的书写方式。再者,如此编程风格非常容易引起不必要的歧义,造成程序的可读性变差。

题 1.16

请写出如下代码的运行结果:

```
int main()
{
    int a,b,c,d;
    a=0;
    b=1;
    c=2;
    d=3;
    printf("%d",a+++b+c+++d++);
    return 0;
}
```

【出处】中兴。

【答案】6。