

## 《高级语言程序设计》考试大纲

### I 考试性质与目的

本科插班生考试是由专科毕业生参加的选拔性考试，我院将根据考生的成绩，按已确定的招生计划，德、智、体全面衡量，择优录取。考试应有较高的信度、效度、必要的区分度和适当的难度。

### II 考试内容

#### 一、考试基本要求

要求考生理解和掌握本科目的基本概念、基本原理和基本方法，能运用本科目知识进行分析，具备分析问题和解决问题的能力的基本能力。

#### 二、考核知识点及考核要求

本大纲的考核要求分为“识记”、“理解”、“应用”三个层次，具体含义是：

1. 识记：能理解有关的概念、知识的含义，并能正确认识和表述。
2. 理解：在识记的基础上，能全面把握基本概念、基本原理、基本方法，能掌握有关概念、原理、方法的区别与联系。
3. 应用：在理解的基础上，能运用基本概念、基本原理、基本方法分析和能解决有关的理论问题和实际问题。

#### 第一章 C 语言概述

##### 一、考核知识点

- 1、C 语言的特点。
- 2、C 程序的结构。
- 3、C 程序的上机步骤。

##### 二、考核要求

- 1、识记：
  - ①C 语言的特点；
  - ②C 程序的结构。
- 2、理解：
  - ①C 程序的上机步骤。

#### 第二章算法

##### 一、考核知识点

- 1、算法的概念。
- 2、算法的特性。
- 3、算法的表示。

##### 二、考核要求

- 1、识记：
  - ①算法的概念；
  - ②算法的特性。
- 2、理解：
  - ①算法的表示。

### 第三章数据类型、运算符与表达式

#### 一、考核知识点

- 1、常量与变量。
- 2、各种数据类型。
- 3、各种数据类型间的混合运算。
- 4、算术运算符与算术表达式。
- 5、赋值运算符与赋值表达式。
- 6、逗号运算符与逗号表达式。

#### 二、考核要求

- 1、识记：
  - 1、常量与变量。
    - 知道什么是常量和变量；
    - 知道常量的几种形式；
    - 知道常量和变量的定义。
  - 2、理解：
    - 2、各种数据类型。
      - 整型数据的表示方法；
      - 整型变量的定义和使用；
      - 实型常量的表示方法；
      - 实型变量的定义与使用方法；
      - 字符常量的表示与转义字符的使用；
      - 字符变量的定义与字符串常量的表示。
    - 3、各种数据类型间的混合运算。
  - 3、应用：
    - 4、算术运算符与算术表达式。
      - 基本算术运算符的应用，特别是“/”与“%”的应用；
      - 正确理解自加与自减运算。
    - 5、赋值运算符与赋值表达式。
      - 掌握赋值运算时类型转换。



启航专插本  
[www.qihangzcb.com](http://www.qihangzcb.com)

- 掌握复合运算符的概念。
- 6、逗号运算符与逗号表达式。

#### 第四章顺序结构程序设计

##### 一、考核知识点

- 1、语句的作用及总体分类。
- 2、赋值语句。
- 3、字符数据的输入/输出。
- 4、格式输入与输出。

##### 二、考核要求

###### 1、识记：

###### 2、理解：

###### 1、语句的作用及总体分类。

知道语句在程序中的作用。

知道 C 语言的语句可以分成哪几类。

###### 3、应用：

###### 2、赋值语句。

熟知赋值语句的一般形式及其含义。

熟练地运用赋值语句来表达实际问题。

能识别不正确的赋值语句。

###### 3、字符数据的输入/输出。

熟知字符数据输入/输出函数的一般形式及其基本用法。

能根据实际情况写出所需的字符输入/输出函数。

###### 4、格式输入与输出。

熟练掌握 printf 函数的格式。

熟练掌握 scanf 函数的格式。

能根据实际情况写出顺序结构的程序设计。

#### 第五章选择结构程序设计

##### 一、考核知识点

- 1、关系运算与关系表达式。
- 2、逻辑运算符与逻辑表达式。
- 3、if 语句。
- 4、switch 语句。

##### 二、考核要求

1、识记：

2、理解：

3、应用：

1、关系运算与关系表达式。

2、逻辑运算符与逻辑表达式。

3、if 语句。

熟知 if 语句的一般形式及其含义。

能用 if 语句解决实际问题。

能识别不正确的 if 语句。

能正确理解 if 语句的嵌套。

4、switch 语句。

熟知 switch 语句的一般形式及其含义。

能用 switch 语句解决实际问题。

能根据实际情况，灵活选用 if 语句和 switch 语句。

注意 if 语句、switch 语句、条件运算符的区别。

#### 第六章循环控制

#### 一、考核知识点

1、while 语句。

2、do - while 语句。

3、for 语句。

4、循环的嵌套

5、break 语句与 continue 语句。

#### 二、考核要求

1、识记：

熟知三种不同循环语句的一般形式及其含义。

2、理解：

能领会三种不同的循环语句的不同用法及它们之间的区别。

领会 break 语句和 continue 语句的用法。

3、应用：

循环语句的使用。

能根据实际情况，灵活地选用三种语句中的一种。

会把三种循环语句嵌套使用。

#### 第七章数组

#### 一、考核知识点

- 1、一维数组的定义和引用。
- 2、多维数组的定义和引用。
- 3、字符数组。
- 4、字符串处理函数。

## 二、考核要求

### 1、识记：

### 2、理解：

#### 1、一维数组的定义和引用。

知道一维数组类型的特点。

熟知一维数组类型的一般形式。

能写出正确的数组变量说明。

### 3、应用：

#### 2、下标变量。

知道什么是下标变量以及它的一般形式。

会确定下标变量的类型。

清楚下标表达式的类型与下标类型的一致性问题的。

对给定数组能写出正确的下标变量。

#### 3、一维数组的应用。

能对数组进行查找、排序、检索等操作。

对一个线性表，会编写程序寻找其中的特定元素。

对一个线性表（未排序的），会编写程序将表中元素按从小到大或从大到小的顺序排列出来。

#### 4、多维数组。

清楚多维数组与一维数组的巨别。

熟知二维数组及其下标变量的一般形式。

掌握二维数组的简单应用。

#### 5、字符数组。

熟知字符数组的定义及其表示。

掌握字符数组的输入与输出。

领会字符串的存储。

#### 6、字符串处理函数。

掌握常见的几个字符串处理函数。

会用字符串函数解决实际问题。

## 第八章 函数

### 一、考核知识点

- 1、函数定义的一般形式。
- 2、函数参数和函数的值。
- 3、函数的调用。
- 4、函数的嵌套调用。
- 5、函数的递归调用。
- 6、局部变量和全局变量。

## 二、考核要求

- 1、识记：
- 2、理解：
- 3、应用：

- 1、函数定义的一般形式

清楚函数命名符的作用。

熟知函数命名符的形式及含义。

能正确使用函数命名符。

- 2、函数参数和函数的值

清楚函数说明的作用。

熟知函数说明的一般形式。

弄清形式参数（值传递与地址传递）的概念。

- 3、函数的调用。

函数调用的一般形式。

函数调用的方法。

- 4、函数的嵌套调用。

- 5、函数的递归调用。

- 6、局部变量和全局变量。



启航专插本  
[www.qihangzcb.com](http://www.qihangzcb.com)

## 第九章预处理命令

### 一、考核知识点

- 1、宏定义。
- 2、“文件包含”处理。

### 二、考核要求

- 1、识记：
- 2、理解：
- 3、应用：

- 1、宏定义。

不带参数的宏定义的使用

带参数的宏定义的使用。

2、“文件包含”处理。

## 第十章 指针

### 一、考核知识点

1、地址和指针的概念。

2、指针变量的概念。

3、数组指针。

4、字符串指针。

### 二、考核要求

1、识记：

2、理解：

1、地址和指针的概念。

3、应用：

2、指针变量的概念。

定义指针变量的方法。

指针变量的引用。

指针变量作为函数参数。

灵活用指针变量编写程序。

3、数组指针。

指向数组元素的指针。

通过指针引用数组元素。

数组名作函数参数。

4、字符串指针。

字符串的表示形式。

字符串指针变量的使用方法。



启航专插本  
[www.qihangzcb.com](http://www.qihangzcb.com)

## 第十一章 结构体与共用体

### 一、考核知识点

1、定义结构体类型变量的方法。

2、结构体变量的引用。

3、结构体数组。

4、共用体的概念。

5、共用体变量的引用。

### 二、考核要求

1、识记：

2、理解：

1、定义结构体类型变量的方法。

把握结构体类型的主要特征。

熟知结构体类型的一般形式。

能写出正确的结构体类型以描述实际问题。

4、共用体的概念。

熟知共用体的概念。

共用体类型的特点。

3、应用：

2、结构体变量的引用。

会用结构体类型说明结构体变量。

掌握结构体变量的引用方法。

3、结构体数组。

会定义结构体数组。

会用结构体数组编写简单程序。

5、共用体变量的引用。



第十二章位运算

一、考核知识点

1、位运算符的含义。

2、简单的位运算。

二、考核要求

1、识记：

2、理解：

清楚各种位运算符的含义。

3、应用：

能作简单的位运算。

启航专插本  
[www.qihangzcb.com](http://www.qihangzcb.com)

第十三章文件

一、考核知识点

1、文件类型指针。

2、文件的打开与关闭。

3、文件的读写。

4、文件的定位。

二、考核要求

1、识记：



## 2、理解：

### 1、文件类型指针。

文件的基本概念。

文件类型指针的定义。

## 3、应用：

### 2、文件的打开与关闭。

清楚 fopen() 函数的功能。

清楚 fclose() 函数的功能。

### 3、文件的读写。

fputc() 函数的功能。

fgetc() 函数的功能。

fputc() 函数与 fgetc() 函数的简单应用。

### 4、文件的定位。

## III 考核形式及试卷结构

1. 本科目考试采用闭卷笔试方式，考试时间为 120 分钟，全卷满分 100 分。

2. 试卷中各部分的占分比例是：第一章约占 5%，第二章约占 5%，第三章约占 5%，第四章约占 5%，第五章约占 10%，第六章约占 15%，第七章约占 15%，第八章约占 5%，第九章约占 5%，第十章约占 15%，第十一章约占 5%，第十二章约占 5%，第十三章约占 5%。

3. 试题对不同能力层次要求的分数比例，一般识记占 20%，理解占 40%，应用占 40%。

4. 试题难易占分比例是：易约占 30%，中约占 50%，难约占 20%。

5. 本科目考试的题型有：填空、选择、阅读程序并计算结果、阅读程序并填空、编程等五种题型（各种题型的具体样式可以参见本纲附录《题型示例》）。根据考核的要求，适当安排各种题型数量的比例，达到考生对知识点的识记、理解、应用的水平和能力。

## IV 参考书目

《C 语言程序设计》（第四版），谭浩强编，清华大学出版社，2010 年。

## V 题型示例

### 一、填空题

1. 在 C 语言中，字符串的结束标记是或。

### 二、选择题

1. 以下能正确定义一维数组的选项是

A) int a[5]={0, 1, 2, 3, 4, 5};

B) char a[]={0, 1, 2, 3, 4, 5};

C) char a={'A', 'B', 'C'};

D) int a[5]="0123";

### 三、程序或程序段，给出运行结果

1. 请读程序：

```
#include 'stdio.h'
main()
{
float x, y;
scanf ("%f", &x);
if(x<0.0)
y=0.0
elseif((x<5.0)&&(x!=2.0))
y=1.0/(x+2.0);
elseif(x<10.0)
y=1.0/x;
else
y=10.0;
printf ("%f\n", y);
}
```

若运行时从键盘上输入 2.0 及回车，请写出上面程序的输出结果。

#### 四、读程序或程序段，并填空

1. 以下函数 fun 用于求两个整数 a 和 b 的最大公约数，请填空。

```
fun(a, b)
inta, b;
{
inti, j, m, n;
if(a>b)
{m=a; a=b; b=m;}
i=a; j=b;
while((n=【1】)!=0)
{j=i; i=n;}
return(i);
}
```

#### 五、编程题

1. 请编写一递归函数 age()，要求以递归方法计算学生的年龄，已知第一位学生年龄最小，为 10 岁，其余学生一个比一个大 2 岁，函数应可返回第 n 位学生的年龄。

递归公式如下：

age(1)=10 (n=1)

$$\text{age}(n) = \text{age}(n-1) + 2(n > 1)$$

## 《数据结构》考试大纲

### I 考试的性质与目的

本科插班生考试是由专科毕业生参加的选拔性考试。《数据结构》是计算机科学与技术专业（本科）的一门专业基础课程，考试主要检查考生对常用基本数据结构（顺序表、链表、栈、队列、树、二叉树、图等）的存储组织、维护操作、基本应用，以及查找、排序等基本算法的掌握程度，以保证后续课程的学习。

### II 考试的内容

#### 一、考试基本要求

##### 1、基本理论知识

- (1)、数据结构的基本概念和基本术语，算法的描述方法和算法分析的基本概念。
- (2)、线性表的基本概念、线性表的基本操作以及这些操作分别在顺序存储和链式存储结构下的实现及复杂度分析。
- (3)、栈和队列的定义、存储结构、实现和典型应用。
- (4)、串的定义及其基本操作。
- (5)、数组的定义、运算和存储。
- (6)、树的定义、基本术语和存储结构，二叉树的定义和性质、二叉树的存储结构及其各种操作，哈夫曼树的概念和应用。
- (7)、图的定义和术语、图的存储结构及其基本操作。
- (8)、各种查找方法的算法、适用范围及时间复杂度的分析。
- (9)、多种内排算法的基本思想和算法的时间复杂度分析，不同排序方法的比较。

##### 2、基本技能

- (1)、能用基本数据结构及其算法描述、解决实际较为简单的问题。
- (2)、能阅读“类C”语言编写的算法，能根据要求用“类C”语言编写算法。
- (3)、能分析算法所完成的功能、运行结果和时间复杂度。

#### 二、考核知识点及考核要求

##### 第一章绪论

##### 一、考核知识点

1. 数据、数据元素、数据项、数据对象、数据结构、逻辑结构、物理结构、元素、结点等基本概念。抽象数据类型的定义、表示和实现方法。
2. 算法、算法的特性、如何用类C语言来描述算法。
3. 算法设计的基本要求以及计算语句频度和估算算法时间复杂度的方法。

##### 二、考核要求

1. 识记：有关数据结构的基本概念，四种基本数据结构的特点。
2. 理解：四种基本数据结构的基本运算，算法复杂度度量的基本概念。
3. 应用：用类 C 语言描述算法

## 第二章线性表

### 一、考核知识点

1. 线性表的定义和基本操作。
2. 线性表顺序存储结构的表示和基本运算。
3. 线性表链式存储，带有附加表头结点和不带附加表头结点的单链表、循环链表和双向链表的表示和查找、插入、删除等基本操作。

### 二、考核要求

1. 识记：线性表基本概念、基本运算，各种链表的表示。 TM
2. 理解：顺序存储和链式存储的比较，各种链表的基本操作算法

## 第三章栈和队列

### 一、考核知识点

1. 栈的定义、操作特点，栈的顺序存储、链式存储以及入栈、出栈、判空（满）、取栈顶元素等基本操作。
2. 队列的定义、操作特点，循环队列的存储、入列、出列、判空（满）、取队首元素等操作。
3. 栈的应用举例，如：数制转换、表达式求值等。

### 二、考核要求

1. 识记：栈和队列的概念、功能、操作特点、主要运算。
2. 理解：栈和队列与一般线性表对比的特殊性，栈和队列的顺序存储和链式存储，循环队列。
3. 应用：栈和队列的常见的使用场合。

## 第四章串

### 一、考核知识点

1. 串的定义、空串的概念。
2. 串的基本操作。
3. 串的顺序存储结构及在顺序存储结构下基本操作的实现。
4. 串的基本模式匹配算法。

### 二、考核要求

1. 识记：串的有关概念。
2. 理解：串的基本操作，串的顺序存储结构及其基本操作。
3. 应用：串的基本操作函数的使用。

## 第五章数组和广义表

### 一、考核知识点

1. 数组的顺序存储结构。
2. 二维数组的按行存储及按列存储和计算数组元素的地址计算公式。
3. 三元组表的概念和基本操作。
4. 广义表的定义。

### 二、考核要求

1. 识记：数组的顺序存储结构，广义表的定义。
2. 理解：二维数组的地址计算，三元组表的表示。
3. 应用：用三元组表解决稀疏矩阵的存储问题。

## 第六章树和二叉树 TM

### 一、考核知识点

1. 树的定义和术语。
2. 二叉树(完全二叉树、满二叉树)的定义和性质、二叉树的存储结构(顺序表示法和二叉链表表示法)。
3. 二叉树遍历算法(先序、中序、后序、层次)。
4. 树和森林转换为二叉树的方法(孩子兄弟表示法)。
5. 树的路径长度、树的带权路径长度、Huffman 树的构造方法。

### 二、考核要求

1. 识记：树的基本概念
2. 理解：二叉树的存储结构、遍历算法，孩子兄弟表示法，树的路径长度，哈夫曼树的构造方法
3. 应用：利用哈夫曼树解决一些最优化问题

## 第七章图

### 一、考核知识点

1. 图的定义。
2. 图的基本术语。
  - (1) 图及无向图、有向图、网、子图、连通图、强连通图。
  - (2) 顶点的度、入度、出度。
  - (3) 顶点间路径、路径长度、环。
3. 图的存储结构
  - (1) 邻接矩阵(2) 邻接表(含逆邻接表)
4. 遍历图
  - (1) 深度优先搜索遍历图的算法及其时间复杂度。
  - (2) 广度优先搜索遍历图的思想及其时间复杂度。
5. 生成树、最小生成树的概念。

- 6. 拓扑排序的方法
- 7. 求最短路径的算法。

## 二、考核要求

- 1. 识记：图的基本概念和术语，最小生成树、拓扑排序、最短路径的概念和算法思想。
- 2. 理解：图的存储方式和基于该存储方式的基本操作（求入度、出度、下一条边等）
- 3. 应用：拓扑序列的应用，最短路径的应用。

## 第八章动态存储管理（不要求）

## 第九章查找

### 一、考核知识点

- 1. 查找、关键字、平均查找长度等概念。
- 2. 静态查找表的查找算法及其效率(最坏和平均查找长度)。  
(1) 顺序查找 (2) 折半查找 (3) 分块查找
- 3. 动态查找表  
二叉排序树定义、构造过程及其查找算法和效率。
- 4. 哈希表  
(1) 哈希表的特点。(2) 构造哈希函数的方法（除留余数法等）。(3) 处理冲突的方法。

### 二、考核要求

- 1. 识记：查找的基本概念，静态查找表和动态查找表的概念，哈希表的概念
- 2. 理解：各种静态查找算法的比较次数分析；二叉排序树的生成过程和查找算法；哈希函数的选择，冲突处理的方法，散列查找的过程。
- 3. 应用：分析各种查找算法的比较次数。

## 第十章内部排序

### 一、考核知识点

- 1. 排序的目的、分类和排序方法的稳定性的定义。
- 2. 直接插入排序的思想
- 3. 快速排序  
(1) 冒泡排序的算法。(2) 快速排序的思想。
- 4. 选择排序  
(1) 简单的选择排序的算法。  
(2) 堆的定义、堆排序的思想。
- 5. 二路归并排序的思想。

### 二、考核要求

- 1. 识记：直接插入排序、冒泡排序、简单选择排序的思想



2. 理解：快速排序、堆排序、二路归并排序的思想，各种排序方法的稳定性、平均比较次数、平均移动次数的分析。

3. 应用：用类 C 或者 C 语言编写直接插入排序、冒泡排序、简单选择排序等排序算法。

第十一章外部排序（不要求）

第十二章文件（不要求）

### III 考试的形式及试卷结构

1、考试的形式：采用闭卷笔试的形式。考试时间 120 分钟，全卷 100 分。

2、试卷中各章所占的比例：第一章约占 8%，第二、三、四、五章共约占 40%，第六章约占 20%，第七章约占 15%，第九章约占 12%，第十章约占 5%。

3、试题对不同能力层次要求的分数比例：识记约占 30%，理解约占 40%，应用约占 30%。

4、试题难易占分比例：易约占 30%，中约占 50%，难约占 20%。TM

5、考卷的结构：试题分为客观题和主观题。客观题一般有填空题、选择题、名词解释、程序填空题等类型；主观题一般有简答题、算法设计题等类型。

### IV 参考书目

主要参考书：《数据结构》（C 语言版）严蔚敏吴伟民编著，清华大学出版社。

### V 题型示例

#### 一、填空题

1、一棵深度为 8（根的层次号为 1）的满二叉树有\_\_\_\_\_个叶子结点。

2、串的长度是指\_\_\_\_\_。

#### 二、选择题

1、一个栈的入栈序列是 a, b, c, d, e, 则栈的不可能的输出序列是\_\_\_\_\_

A. edcba B. decba C. dceab D. abcde

2、对于栈操作数据的原则是\_\_\_\_\_。

A. 先进先出 B. 后进先出 C. 后进后出 D. 不分顺序

#### 三、名词解释

1、连通图

2、完全二叉树

#### 四、程序填空题

下面的程序段是在一棵二叉排序树中查找给定的关键字，找到返回 1，找不到返回 0。请把该程序补充完整。

二叉树结点定义如下：

```
struct Tree{
    ElemType data; /* 存放数据 */
    struct Tree *left; /* 指向左子树 */
    struct Tree *right; /* 指向右子树 */
```

元素类型 ElemType 定义如下：

```
struct ElemType{
    KeyType key; /*关键字 */
    ... /*其他数据项*/
```

```
intFind(Tree*boot, ElemTypeitem)
{
    Tree*p=boot;
    while(_____) {
        if(item.key<boot->data.key)
            _____
        elseif(item.key>p->data.key)
            _____
        else
            _____
    }
    return(0);
}
```

### 五、简答题

- 1、试比较链式存储和顺序存储的优缺点。
- 2、已知一棵二叉树的中序序列和后序序列分别为 BDCEAFHG 和 DECBHGFA，试写出其先序序列。

### 六、算法设计题

设计一算法，实现将一个递减的数组 A[0..n-1] 和一个带头结点的递增单链表 B 合并成一个带头结点的递增链表 C。已知单链表的数据结构定义为：

```
structSingleLink{
    ElemTypedata;
    SingleLink*next;
};
```

请用函数原型：

```
SingleLink*LinkAAndB(ElemTypeA[], intn, SingleLink*B);
```

数组 A 和要链接的单链表 B 通过函数参数传递，n 是数组的规模。函数返回值是生成的链表。