

目录

I 考查目标	2
II 考试形式和试卷结构.....	2
III 考查内容	2
IV. 题型示例及参考答案.....	7

全国硕士研究生入学统一考试 程序设计考试大纲

I 考查目标

全国硕士研究生入学统一考试计算机技术、软件工程专业学位硕士《程序设计》考试是为江苏大学招收以上硕士生设置的具有选拔性质的考试科目。其目的是科学、公平、有效地测试考生是否具备攻读计算机技术、软件工程专业学位硕士所必须的基本素质、一般能力和培养潜能,以利于选拔具有发展潜力的优秀人才入学,为国家的经济建设培养具有良好职业道德、法制观念和国际视野、具有较强分析与解决实际问题能力的专业人才。考试要求考生比较系统地掌握程序设计的基本理论、基本算法和方法,能够运用程序设计的基本方法和技巧分析、判断和设计有关理论与实际问题的 C 语言或 C++ 语言程序。

II 考试形式和试卷结构

一、试卷满分及考试时间

试卷满分为 150 分,考试时间 180 分钟。

二、答题方式

答题方式为闭卷、笔试。

三、试卷内容与题型结构

单项选择题	10 题,每小题 2 分, 共 20 分
填空题	题数不定,每空 2 分, 共 20 分
阅读程序并写出结果	5 题,每小题 4 分, 共 20 分
程序填空	题数不定, 每空 3 分, 共 45 分
编程题	3 题,每小题 15 分, 共 45 分

III 考查内容

1 程序设计和 C 语言

1.1 什么是计算机程序

1.2 什么是计算机语言

1.3 C 语言的发展及其特点

1.4 最简单的 C 语言程序

1.4.1 最简单的 C 语言程序举例

1.4.2 C 语言程序的结构

1.5 运行 C 程序的步骤与方法

1.6 程序设计的任务

2 算法——程序的灵魂

2.1 什么是算法

2.2 简单的算法举例

2.3 算法的特性

2.4 怎样表示一个算法

2.4.1 用自然语言表示算法

2.4.2 用流程图表示算法

2.4.3 三种基本结构和改进的流程图

- 2.4.4 用 N-S 流程图表示算法
- 2.4.5 用伪代码表示算法
- 2.4.6 用计算机语言表示算法
- 2.5 结构化程序设计方法

- 3 最简单的 C 程序设计——顺序程序设计
- 3.1 顺序程序设计举例
- 3.2 数据的表现形式及其运算
- 3.2.1 常量和变量
- 3.2.2 数据类型
- 3.2.3 整型数据
- 3.2.4 字符型数据
- 3.2.5 浮点型数据
- 3.2.6 怎样确定常量的类型
- 3.2.7 运算符和表达式
- 3.3 C 语句
- 3.3.1 C 语句的作用和分类
- 3.3.2 最基本的语句——赋值语句
- 3.4 数据的输入输出
- 3.4.1 输入输出举例
- 3.4.2 有关数据输入输出的概念
- 3.4.3 用 printf 函数输出数据(注: C++可以用 cout)
- 3.4.4 用 scanf 函数输入数据(注: C++可以用 cin)
- 3.4.5 字符数据的输入输出

- 4 选择结构程序设计
- 4.1 选择结构和条件判断
- 4.2 用 if 语句实现选择结构
- 4.2.1 用 if 语句处理选择结构举例
- 4.2.2 if 语句的一般形式
- 4.3 关系运算符和关系表达式
- 4.3.1 关系运算符及其优先次序
- 4.3.2 关系表达式
- 4.4 逻辑运算符和逻辑表达式
- 4.4.1 逻辑运算符及其优先次序
- 4.4.2 逻辑表达式
- 4.4.3 逻辑型变量
- 4.5 条件运算符和条件表达式
- 4.6 选择结构的嵌套
- 4.7 用 switch 语句实现多分支选择结构
- 4.8 选择结构程序综合举例

- 5 循环结构程序设计
- 5.1 为什么需要循环控制

- 5.2 用 while 语句实现循环
- 5.3 用 do...while 语句实现循环
- 5.4 用 for 语句实现循环
- 5.5 循环的嵌套
- 5.6 几种循环的比较
- 5.7 改变循环执行的状态
 - 5.7.1 用 break 语句提前终止循环
 - 5.7.2 用 continue 语句提前结束本次循环
 - 5.7.3 break 语句和 continue 语句的区别
- 5.8 循环程序举例

6 利用数组处理批量数据

- 6.1 怎样定义和引用一维数组
 - 6.1.1 怎样定义一维数组
 - 6.1.2 怎样引用一维数组元素
 - 6.1.3 一维数组的初始化
 - 6.1.4 一维数组程序举例
- 6.2 怎样定义和引用二维数组
 - 6.2.1 怎样定义二维数组
 - 6.2.2 怎样引用二维数组的元素
 - 6.2.3 二维数组的初始化
 - 6.2.4 二维数组程序举例
- 6.3 字符数组
 - 6.3.1 怎样定义字符数组
 - 6.3.2 字符数组的初始化
 - 6.3.3 怎样引用字符数组中的元素
 - 6.3.4 字符串和字符串结束标志
 - 6.3.5 字符数组的输入输出
 - 6.3.6 使用字符串处理函数
 - 6.3.7 字符数组应用举例

7 用函数实现模块化程序设计

- 7.1 为什么要用函数
- 7.2 怎样定义函数
 - 7.2.1 为什么要定义函数
 - 7.2.2 定义函数的方法
- 7.3 调用函数
 - 7.3.1 函数调用的形式
 - 7.3.2 函数调用时的数据传递
 - 7.3.3 函数调用的过程
 - 7.3.4 函数的返回值
- 7.4 对被调用函数的声明和函数原型
- 7.5 函数的嵌套调用
- 7.6 函数的递归调用

- 7.7 数组作为函数参数
 - 7.7.1 数组元素作函数实参
 - 7.7.2 数组名作函数参数
 - 7.7.3 多维数组名作函数参数
- 7.8 局部变量和全局变量
 - 7.8.1 局部变量
 - 7.8.2 全局变量
- 7.9 变量的存储方式和生存期
 - 7.9.1 动态存储方式与静态存储方式
 - 7.9.2 局部变量的存储类别
 - 7.9.3 全局变量的存储类别
 - 7.9.4 存储类别小结
- 7.10 关于变量的声明和定义
- 7.11 内部函数和外部函数
 - 7.11.1 内部函数
 - 7.11.2 外部函数

- 8 善于利用指针
 - 8.1 指针是什么
 - 8.2 指针变量
 - 8.2.1 使用指针变量的例子
 - 8.2.2 怎样定义指针变量
 - 8.2.3 怎样引用指针变量
 - 8.2.4 指针变量作为函数参数
 - 8.3 通过指针引用数组
 - 8.3.1 数组元素的指针
 - 8.3.2 在引用数组元素时指针的运算
 - 8.3.3 通过指针引用数组元素
 - 8.3.4 用数组名作函数参数
 - 8.3.5 通过指针引用多维数组
 - 8.4 通过指针引用字符串
 - 8.4.1 字符串的引用方式
 - 8.4.2 字符指针作函数参数
 - 8.4.3 使用字符指针变量和字符数组的比较
 - 8.5 指向函数的指针
 - 8.5.1 什么是函数指针
 - 8.5.2 用函数指针变量调用函数
 - 8.5.3 怎样定义和使用指向函数的指针变量
 - 8.5.4 用指向函数的指针作函数参数
 - 8.6 返回指针值的函数
 - 8.7 指针数组和多重指针
 - 8.7.1 什么是指针数组
 - 8.7.2 指向指针数据的指针
 - 8.7.3 指针数组作 main 函数的形参

8.8 动态内存分配与指向它的指针变量

8.8.1 什么是内存的动态分配

8.8.2 怎样建立内存的动态分配

8.8.3 void 指针类型

9 用户自己建立数据类型

9.1 定义和使用结构体变量

9.1.1 自己建立结构体类型

9.1.2 定义结构体类型变量

9.1.3 结构体变量的初始化和引用

9.2 使用结构体数组

9.2.1 定义结构体数组

9.2.2 结构体数组的应用举例

9.3 结构体指针

9.3.1 指向结构体变量的指针

9.3.2 指向结构体数组的指针

9.3.3 用结构体变量和结构体变量的指针作函数参数

9.4 用指针处理链表

9.4.1 什么是链表

9.4.2 建立简单的静态链表

9.4.3 建立动态链表

9.4.4 输出链表

9.5 共用体类型

9.5.1 什么是共用体类型

9.5.2 引用共用体变量的方式

9.5.3 共用体类型数据的特点

9.6 使用枚举类型

9.7 用 typedef 声明新类型名

10 对文件的输入输出

10.1 C 文件的有关基本知识

10.1.1 什么是文件

10.1.2 文件名

10.1.3 文件的分类

10.1.4 文件缓冲区

10.1.5 文件类型指针

10.2 打开与关闭文件

10.2.1 用 fopen 函数打开数据文件

10.2.2 用 fclose 函数关闭数据文件

10.3 顺序读写数据文件

10.3.1 怎样向文件读写字符

10.3.2 怎样向文件读写一个字符串

10.3.3 用格式化的方式读写文件

10.3.4 用二进制方式向文件读写一组数据

- 10.4 随机读写数据文件
- 10.4.1 文件位置标记及其定位
- 10.4.2 随机读写
- 10.5 文件读写的出错检测

参考书:

1. 谭浩强.《C 程序设计》(第二版及其以后版本), 清华大学出版社
2. 沈显君等.《C++语言程序设计教程》(第二版), 清华大学出版社, 2010 年

IV. 题型示例及参考答案

一、单项选择题(每小题 2 分,共 20 分)

1. 下面选项中()是不合法的 C 语言用户标识符。
A. sin B. book C. _name D. Example2.1
2. 设有枚举类型定义: enum color{red, yellow, blue=9, white, black}; 则枚举元素 black 的值是()。
A. 4 B. 5 C. 11 D. 12
3. 若 w=1,x=2,y=3,z=4; 则条件表达式 w<x?w:y<z?y:z 的值是 ()。
A. 1 B. 2 C. 3 D. 4
4. 若用数组名作为实参,则传递给形参的是()。
A. 数组首地址 B. 数组第一个元素的值
C. 数组元素的个数 D. 数组中全部元素的值
5. 一个可执行 C(C++)程序的开始执行点是()。
A. 程序中的第一个语句 B. 包含文件的第一个函数
C. 名为 main 的函数 D. 程序中的第一个函数
6. 若 int k,*p; p=&k; 要实现 k 的内容加 5,则下面的语句 ()是错的。
A. k+=5 B. k=*p+5 C. *p=k+5 D. *p=&k+5
7. 若有语句组 typedef char *STRING; STRING a; 则以下叙述中正确的是()。
A. a 是一个新类型 B. a 是一个字符型指针变量
C. a 是一个字符串常量 D. a 是一个字符型变量
8. 下面能正确进行字符串赋值操作的语句是()。
A. char s[5]={"ABCDE"}; B. char s[5]={'A','B','C','D','E'};
C. char *s; s="ABCDEF"; D. char *s; strcpy(s,"ABCDE");
9. 以下函数声明中,存在着语法错误的是()。
A. int AA(int a,int); B. int BB(int,int);
C. void CC(int *, int *); D. void DD(int x, y);
10. C(C++)语言中的文件类型划分为()。
A. 索引文件和文本文件两种 B. ASCII 文件和二进制文件两种
C. 只有文本文件一种 D. 只有二进制文件一种

二、填空题(每空 2 分,共 20 分)

1. 任何算法都可以用顺序结构、_____、_____这三种基本结构及其组合来描述。
2. $\sqrt{\sin^{2.5}(x)}$ 的 C(C++)表达式是_____。
3. $0 \leq x \leq 1$ 的 C(C++)语言的表达式是_____。
4. 有一种只在循环语句或 switch 语句中使用的流程控制语句,表示该语句的关键字是_____。
5. 已有定义: int a,b; 则表达式 a=2,b=5,a++,b++,a+b 的值是_____。
6. 若 w=1,x=2,y=3,z=4; 则条件表达式 w<x?w:y<z?y:z 的值是_____。
7. 已有定义: int x=3,y=4,z=5; 则表达式!(x+y)+z-1&& y+z/2 的值是_____。
8. 与"!a==0"等价的 C(C++)表达式是_____。
9. 在 5 个运算符"* (乘号)、<=、!=、!、&&"中,优先级由高到低的顺序是_____。

三、阅读下列程序,写出程序运行结果(本题有 5 小题,每小题 4 分,共计 20 分)

1. #include <stdio.h>

/*C++: #include <iostream.h> */

```
void main( )
{
    int m, n, r;
    m=75; n=30;
    while ( r=m%n) { m=n; n=r; }
    printf("%d \n", n);
    /*C++: cout<<n<<endl; */
}
```

2. #include <stdio.h>

/*C++: #include <iostream.h> */

```
int fl(int m, int n)
{
    int a,b;
    a=n++; b= --m;
    return (a+b);
}

void main( )
{ int a,b,c;
  a=3; b=5;
  printf("a=%d,b=%d\n",a,b);
  /*C++: cout<<"a="<<a<<",b="<<b<<endl; */
  c=fl(a,b);
  printf("a=%d,b=%d,c=%d\n",a,b,c);
  /*C++: cout<<"a="<<a<<",b="<<b<<","c="<<c <<endl; */
}
```

3. #include <stdio.h>

/*C++: #include <iostream.h> */

```
void main( )
```

```

{
    int s[6][6],j,k;
    for(j=0;j<6;j++)
        for(k=0;k<6;k++)
            *((s+j)+k)=j-k;
    for(j=0;j<6;j++)
    {
        for(k=0;k<6;k++)
            printf("%4d",*((s+j)+k));
        /*C++: cout<<setw(4)<<*((s+j)+k) ;*/
        printf("\n");
        /*C++: cout<<endl;*/
    }
}

```

```

4. #include <stdio.h>
/*C++: #include <iostream.h> */
int find(int *a, int n, int x)
{
    int *p=a+n;
    p--;
    while(*p!=x && p>=a) p--;
    return p-a;
}
void main( )
{
    static int a[]={1,2,3,4,5,6,7,8};
    int k=20;
    printf("%4d",find(a, 8, k));
    /*C++: cout<<setw(4)<< find(a, 8, k);*/
}

```

```

5. #include <stdio.h>
/*C++: #include <iostream.h> */
void main()
{
    int a=1,b=2,c=3;
    ++a;    c += ++b;
    if (c>=4){
        int b=4,c;    c = 3*b;    a += c;
        printf("First:  %d, %d, %d\n",a,b,c);
        /*C++: cout<<" First:  "<<a<<," "<<b<<," "<<c <<endl; */
        a += c;
        printf("Second: %d, %d, %d\n",a,b,c);
        /*C++: cout<<" Second: "<<a<<," "<<b<<," "<<c <<endl; */
    }
    printf("Third:  %d, %d, %d\n",a,b,c);
}

```

```

/*C++: cout<<" Third:  "<<a<<" "<<b<<" "<<c <<endl; */
}

```

四、程序填空(阅读下列程序说明和 C(C++)代码, 将应填入 (n) 处的字句写在答卷的对应栏内, 每空 3 分, 共计 45 分)

1、【程序说明】本程序功能是输入一行字符, 分别统计出其中的英文字母、空格、数字和其他字符的个数。

```

# (1) <stdio.h>
void main()
{ char ch;
  int letter, digit, space, other;
  (2);
  digit=0; space=0; other=0;
  while ((ch=getchar())(3) '\n')
    if (ch>='a' && ch<='z' || ch>='A' && ch<='Z') letter++;
    else if ((4)) digit++;
        else if (ch==' ') (5);
            else other++;
  printf("letter=%4d, digit=%4d, space=%4d, other=%4d\n", letter, digit, space, other);
}

```

2. 【程序说明】本程序功能是求 1 至 N 之间的所有质数(素数)。(设 N>1, 其值由键盘输入)

```

# include <stdio.h>
# include <math.h>
void main()
{
  (6);
  printf("输入一个>1 整数: ");
  do {
    scanf("%d", &N);
  } while ((7));
  printf("1 至 %d 之间的所有质数是: 2", N);
  for (m=3; m<=(8); m++)
  {
    int i, tmp=(int)sqrt(m);
    for (i=2; i<=tmp; i++) if ((9)) break;
    if ((10)) printf(" %d", m);
  }
  printf("\n");
}

```

3. 【程序说明】本函数 creatlist 用来建立一个带头结点的学生数据的单向链表, 新的结点总是插入在链表的末尾。链表的头指针作为函数值返回, 链表最后一个结点的 next 域放入 NULL, 作为链表结束标志。创建中如输入的学号为 0, 则创建结束(学号为 0 不存入链表)。

```

#define NULL 0
#define LEN sizeof(struct student)
struct student
{
    long num;        /* 学号 */
    float score;    /* 成绩 */
    struct student *next;
};
_____ (11) creatlist() /*函数返回链表的起始地址 */
{
    struct student *head, *p1, *p2;

    p1 = p2 = (struct student *)malloc(LEN); /*开辟一个新单元*/
    scanf ("%ld %f", &p1->num, &p1->score);
    head = NULL;
    while(p1->num != 0)
    {
        if(head == NULL) _____ (12) _____;
        else _____ (13) _____;
        p2 = p1;
        p1 = _____ (14) _____;
        scanf ("%ld,%f", &p1->num, &p1->score);
    }
    p2->next = _____ (15) _____;
    free(p1);
    return (head);
}

```

五、用 C(C++)语言编写下列程序（共 45 分）

1（15 分）、用递归函数实现勒让德多项式：

$$P_n(x) = \begin{cases} 1 & n=0 \\ x & n=1 \\ ((2n-1)xP_{n-1}(x) - (n-1)P_{n-2}(x))/n & n>1 \end{cases}$$

编写一程序，输入 n, x，输出 $P_n(x)$ 的值。

2（15 分）、利用牛顿切线公式计算数 a（不为 0）的立方根。根据牛顿公式，其第 n+1 次近似根与第 n 次近似根满足： $x_{n+1} = (2x_n^3 + a) / (3x_n^2)$ 。要求相邻两次近似根的差的绝对值小于等于 10^{-6} 时停止。初次迭代近似根 x_0 取值为 1

3（15 分）、求 2 到 500 之间的所有亲密数对并保存到文件 data.txt 文件中。亲密数对的定义为：如果 M 的因子(包括 1，但不包括自身)之和为 N，且 N 的因子之和为 M，则称 M 与 N 为一对亲密数。

参考答案

一、单项选择题

1-5:DCAAB

6-10:DBCDB

二、填空题

1. 选择结构、循环结构

2. sqrt(pow(sin(x),2.5))

3. (x>=0)&&(x<=1)

4. break

5. 9

6. 1

7. 1

8. a 或 a!=0

9. !、*(乘)、<=、!=、&&

三、阅读下列程序，写出程序运行结果

1.

15

2.

a=3,b=5

a=3,b=5,c=7

3.

0 -1 -2 -3 -4 -5

1 0 -1 -2 -3 -4

2 1 0 -1 -2 -3

3 2 1 0 -1 -2

4 3 2 1 0 -1

5 4 3 2 1 0

4.

-1

5.

First: 14, 4, 12

Second: 26, 4, 12

Third: 26,3, 6

四、程序填空

1. include

2. letter =0

3. !=

4. ch>='0' && ch<='9'

5. space++ 或 ++space 或 space +=1

```

6. int N,m
7. N<=1
9. m%i==0

8. N
10. i>tmp

11. struct student *
13. p2->next=p1
15. NULL

12 head=p1
14. (struct student *)malloc(LLEN)

```

五、用 C(C++)语言编写下列程序 (共 45 分)

1(15 分)、

```

double p(int n, double x)
{
    if(n==0) return 1;
    else if(n==1) return x;
    else if(n>1) return ((2*n-1)*x*p(n-1,x)-(n-1)*p(n-2,x))/n;
    return 0;
}

void main()
{
    int n;
    double x;
    printf("请输入整数 n,实数 x:\n");
    scanf("%d %lf",&n, &x);
    printf("计算结果是%lf:\n", p(n,x));
}

```

2 (15 分)

```

#include <math.h>
#define EPS 1e-6
main()
{
    double x, y=1, a;
    printf("请输入要开方的数。 \n");
    scanf("%lf", &a);
    do {
        x = y; /*保存前次近似根*/
        y=2*x/3+ a/(3*x*x); /*计算本次近似根*/
    } while ( fabs(y-x) > EPS);
    printf("The cubic root of %lf is %10.5lf\n",a,y);
}

```

3 (15 分)、

```
# include <stdio.h>
void main()
{   int m, n,i,k ;
    FILE *fp;

    if ((fp = fopen("data.txt","w")) == NULL)
    {   printf("cannot open file\n");
        exit(0);
    }

    for(m=2; m<=500; m++){
        n=1;
        for (i=2; i<=m/2; i++)      if ( m%i==0 )  n=n+i;
        k= 1;
        for (i=2; i<=n/2; i++)      if (n%i==0)  k=k+i;
        if( k==m && m<=k) {
            printf("m=%3d, n=%3d\n", m, n);
            fprintf(fp,"m=%3d, n=%3d\n", m, n);
        }
    }
    if(fclose(fp)!=0)    printf("file close error\n");
}
```