

## C 语言程序设计（下）期末复习题

### 一、单项选择题（60 分）

- 1 C 语言的程序一定从 C 开始执行。  
A) 程序的第一条语句      B) 程序的第一个函数  
C) 程序中的 **main** 函数      D) 文件包含中的第一个文件
- 2 下面叙述中正确的是 B。  
A) C 程序的最小功能单位是源程序      B) C 程序的最小功能单位是函数  
C) C 程序的最小功能单位是语句      D) C 程序的最小功能单位是表达式
- 3 C 语言中函数返回值的类型是有 A 决定的。  
A) 函数定义时的类型      B) **return** 表达式 语句中表达式的类型  
C) 必须是整型      D) 函数调用时主调函数的类型
- 4 如果定义函数为 `int *fun( )`, 则函数 **fun** 的返回值为 B。  
A) 整型变量      B) 一个指向整型变量的指针  
C) 一个整型函数的入口地址      D) 一个指向整型函数的指针
- 5 关于函数的叙述中正确的是 C。  
A) 函数只能被 **main** 函数调用      B) 每个函数调用之后都必须有返回值  
C) 函数可以嵌套调用      D) 函数可以嵌套定义
- 6 如果有函数定义形式如下：  
`Int fun( int *p1, int *p2)`  
`{.....}`  
如果在主函数中已经定义 `int a,b,c[10], d[10], *p,*q;` 则错误的函数调用形式是 D。  
A) `fun(&a, &b);`      B) `fun(c,d);`  
C) `fun(p,q);`      D) `fun(a,b);`
- 7 如果定义了 `ina l,a[10], *pa=a;` 则数组中的下标为 *i* 的元素不可以表示为 C。  
A) `a[i]`      B) `*(a+i)`  
C) `pa+l`      D) `* (pa+i)`
- 8 如果定义了 `int l, a[10], *pa=a;` 则为数组 **a** 中元素输入数据的正确语句是 D。  
A) `scanf("%d", a[i]);`      B) `scanf("%d", *pa[i]);`  
C) `scanf("%d", *(a+i));`      D) `scanf("%d", pa+i);`
- 9 函数调用的语句是 `fun( a,b,(c,d),(e,f,g))`; 该函数参数的个数为 A。  
A) 4      B) 5  
C) 6      D) 7
10. 若在 C 语言中定义函数时没有定义类型, 则系统默认该函数的类型是 C。

- A) float                      B) long  
C) int                         D) void

11 如果一个函数位于 C 程序文件的上部，在该函数体内说明语句后的复合语句中定义了一个变量，则该变量   C  。

- A) 为全局变量，在本程序文件范围内有效  
B) 为局部变量，只在该函数内有效  
C) 为局部变量，只在该复合语句中有效  
D) 定义无效，为非法变量

12 如有以下程序段,运行之后的输出结果为   B  。

```
int a=10;
int main()
{ int fun();
  int a=1;
  printf("%5d", a);
  { int a=2;
    printf("%5d", a);
  }
  fun();
}
int fun()
{ printf("%5d",a);
}
```

- A) 10    1    2                      B) 1    2    10  
C) 1    2    不确定                 D) 1    2    1

13 关于以下 fun 函数的功能叙述中，正确的是   A  。

```
int fun(char *s)
{
  char *t=s;
  while(*t++) ;
  t--;
  return(t-s);
}
```

- A) 求字符串 s 的长度                B) 比较两个串的大小  
C) 将串 s 复制到串 t                 D) 求字符串 s 所占字节数

14 下面的语句中，正确的初始化语句是   D  。

- A) char str[5]="CHINA";  
B) char \*str=CHINA;  
C) char str[6], str="CHINA";  
D) char \*str; str="CHINA";

15 下列程序运行后的结果为   D   。

```
int main()
{
    int a = 10;
    int *p,**q;
    p = &a;
    q = &p;
    printf("%d,%d",*p,*q);
}
```

- A) 10,0      B) 0,10      C) 10,10      D) 以上结果都不对

16 若有以下说明和语句，且 $0 \leq i < 10$ ，则对数组元素地址的正确表示是   D  

```
int a[]={1,2,3,4,5,6,7,8,9,0}, *p,i;
p=a;
```

- A) &(a+1)    B) a++    C) &p    D) &p[i]

17 若已经定义 `int s[4][5], (*ps)[5]; ps=s;` 则对 s 数组元素的正确引用形式是   C  

- A) ps+1    B) \*(ps+3)    C) ps[0][2]    D) \*(ps+1)+3

18 下列程序运行后，输出的结果是   D   。

```
int main()
{
    int a[9][4] = {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16};
    printf("%d\n",**(a+8));
}
```

- A) 8      B) 9      C) 10      D) 0

19 如果已经定义 `int c[4][5], (*cp)[5]; cp=c;` 则对 C 数组元素的正确引用是   D   。

- A) cp+1    B) \*(cp+3)    C) \*(cp+1)+3    D) \*(\*cp+2)

20 根据以下的定义，能输出字母 M 的语句是   D  

```
struct person {char name[9];int age; }
struct person class[10]={ "John",17,"Paul",19, "Mary",18,
"Adam",16, };
```

- A) printf("%c\n",class[3].name);  
B) printf("%c\n",class[3].name[1]);  
C) printf("%c\n",class[2].name[1]);  
D) printf("%c\n",class[2].name[0]);

21 若有以下说明和语句，则值为 6 的表达式是   D  

```
struct st
```

```

    { int n;
      struct st *next;
    };
    struct st a[3], *p;
    a[0].n=5; a[0].next=&a[1];
    a[1].n=7; a[1].next=&a[2];
    a[2].n=9; a[0].next='\0';
    p=&a[0];

```

- A) p++->n    B) p->n++    C) (\*p).n++    D) ++p->n

22 设有如下定义:

```

struct sk
{int a;float b;}data, *p;

```

当 p=&data; , 则对 data 中的 a 域的正确引用是 B 。

- A) (\*p).data.a    B) (\*p).a    C) p->data.a    D) p.data.a

23 以下关于结构体的叙述中**错误**的是 D

- A) 仅结构体类型定义之后, 编译系统不为其分配内存;  
 B) 结构体变量定义之后, 编译系统为结构体变量分配内存;  
 C) 可以通过指向结构体变量的指针访问结构体成员;  
 D) 结构体类型不可以嵌套。

24 C 语言中系统的标准数入文件是指 A

- A) 键盘    B) 显示器    C) 硬盘    D) 硬盘

25 有以下程序段, 以下选项中表达式的值为 11 的是 C

```

struct st
{ int x; int *y;}*pt;
int a[]={1,2};b[]={3,4};
struct st c[2]={10,a,20,b};
pt=c;

```

- A) \*pt->y    B) pt->x    C) ++pt->x    D) (pt++)->x

26 已经定义函数如下:

```

void fun ( char *w, int m )
{ char s, *p1, *p2 ;
  p1=w; p2=w+m-1;
  while(p1<p2)
  { s=*p1++; *p1=*p2--; *p2=s; }
}

```

如果有数组 a[10]="ABCDEFGH";当调用函数 fun ( a , strlen(a) );后, 数组 a 的内容为 B .

- A) GFEDCBA                      B) AGAAGAG  
C) AGAGGAG                      D) GAGGAGA

27 要打开一个已存在的非空文件“file”用于修改，选择正确的语句是   D  。

- A) fp=fopen("file", "r");      B) fp=fopen("file", "a+");  
C) fp=fopen("file", "w");      D) fp=fopen("file", "r+");

28 库函数 fgets(s, n, f) 的功能是   B  

- A) 从文件 f 中读取长度为 n 的字符串存入指针 s 所指的内存  
B) 从文件 f 中读取长度不超过 n-1 的字符串存入指针 s 所指的内存  
C) 从文件 f 中读取 n 个字符串存入指针 s 所指的内存  
D) 从文件 f 中读取长度为 n-1 的字符串存入指针 s 所指的内存

29 为了改变文件的位置指针，应当使用的函数是   A  。

- A) fseek( )    B) rewind( )      C) ftell( )      D) feof( )

30 若 fp 是指向某文件的指针，当指针已到文件末尾，则库函数 feof(fp) 的返回值是   C  

- A) EOF                      B) -1                      C) 非零值                      D) NULL

### 程序题

1、（10 分，每个用例 5 分）给定一个区间，输出其中的对称的素数

例如：输入：10 20

输出：11

测试用例 1：输入 100 150

输出：101 131

测试用例 2：输入 900 1000

输出：919 929

#include <stdio.h>

using namespace std;

bool isPrime(int n)

{

    if(n==1) return false;

    if(n!=2&& n%2==0) return false;

    for(int i=3; i\*i<=n; i=i+2)

    {

```

        if(n%i==0) return false;

    }

    return true;

}

bool isSym(int n)

{

    if(n<12&& n!=10) return true;

    if(n>100&& n<1000&& n/100==n%10) return true;

    if(n>1000&& n<10000&& n/1000==n%10&& n%1000/100==n%100/10)
        return true;

    return false;

}

int main(int argc,char * argv[])

{

    int a , b;
    scanf("%d %d", &a, &b);
    for(int i = a; i <=b; i++)
    {

        if( isSym(i)&& isPrime(i) )
        {
            printf("%d",i);

        }

    }

    return 0;

}

```

2、（15 分，每个用例 5 分）输入数字和字母，输出由字母组成的 X 图形。

例如: 3 B

```
D  D
  C C
    B
  C C
D  D
```

测试用例 1: 3 g

```
i  i
  h h
    g
  h h
i  i
```

测试用例 2: 2 D

```
E E
  D
E E
```

测试用例 3: 1 b

```
b
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
main()
```

```
{
    int i,j,n;
    char ch;
    printf("\nPlease enter n & letter:");
    scanf("%d %c",&n,&ch);
    for(i=0;i<=2*(n-1);i++)
    {
        for(j=0; j<=2*(n-1); j++)
        {
            if(j==2*(n-1)-i || i == j)
                printf("%c",ch+abs((n-1) - i));
            else
                printf(" ");

        }
        printf("\n");
    }
}
```

3、（15 分，每个用例 5 分）给定一个字符串，输出不含重复字符的最长子串，如有并列则全列出来

例如：输入：good morning

输出：d morni

测试用例 1：输入：where are you

输出：are you

测试用例 2：输入 good

输出：go

od

测试用例 3：输入：thank you

输出：thank you

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main()
```

```
{
```

```
    int iStart,iLen,strlen[256];
```

```
    int stringLenth;
```

```
    int maxLenth = 0;
```

```
    char str[256];
```

```
    gets(str);
```

```
    stringLenth = strlen(str);
```

```
    for(iStart = 0; iStart < stringLenth; iStart++)
```

```
    {
```

```
        int isStop = 0;
```

```
        for(iLen = 1; iLen < stringLenth - iStart; iLen++)
```

```
        {
```

```
            for(int isearch = iStart; isearch < iStart + iLen; isearch++)
```

```
            {
```

```
                if( str[isearch] == str[iStart + iLen] )
```

```
                {
```

```
                    isStop = 1;
```

```
                    break;
```

```
                }
```

```
            }
```

```
            if(isStop)
```

```
                break;
```

```
        }
```

```
        strlen[iStart] = iLen;
```

```
        if(maxLenth < iLen)
```

```
            maxLenth = iLen;
```

```
    }
```

```
    for(iStart = 0; iStart < stringLenth; iStart++)
```

```
{
```



```
        if(strlen[iStart] == maxLenth)
        {
            for(iLen = 0; iLen < maxLenth; iLen++)
                printf("%c",str[iStart+iLen]);
            printf("\n");
        }
    }

    //printf("%s",substr);
    getchar();
    getchar();

    return 0;
}
```