

# C 语言基础测试题目---理论

班级：\_\_\_\_\_

姓名：\_\_\_\_\_

成绩：\_\_\_\_\_

1. 以下不能对一维数组 a 进行正确初始化的语句是\_\_\_\_\_。

A. int a[10]=(0,0,0,0)

B. int a[10]={};

C. int a[]={0};

D. int a[10]={10\*1};

2. 若执行以下程序段后，则 n 的值是 (\_\_\_\_\_)

```
int a=1, b=2, m=2, n=2;
```

```
(m = a > b) && ++n;
```

A、 1

B、 2

C、 3

D、 4

3. 以下程序的运行结果是 (\_\_\_\_\_ )。

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

```
int main(void)
```

```
{
```

```
    int m = 5;
```

```
    if(m++ > 5)
```

```
        printf("%d\n", m);
```

```
    else
```

```
        printf("%d\n", m++);
```

```
}
```

A、 7

B、 6

C、 5

D、 4

4. 下列程序执行的两次打印结果是 (1) \_\_\_\_\_ (2) \_\_\_\_\_

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

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

```
{
```

```
    int a;
```

```
    int *p;
```

```
    p = &a;
```

```
    *p = 0x500;
```

```
    a = (int )(*(&p));
```

```
    if(a == (int)p)
```

```
        printf("equal!\n");
```

```
    else
```

```
        printf("not equal!\n");
```

```
    a = (int )(&(*p));
```

```
    if(a == (int)p)
```

```
        printf("equal!\n");
```

```
    else
```

```
        printf("not equal!\n");
```

```
    return 0;
```

```
}
```

5. 在 scanf()函数的格式说明符中，长整型数的输入格式符是\_\_\_\_\_。

A. %lf

B. %ld

C. %d

D. %x

6. 下面程序的输出结果是\_\_\_\_\_

```
int main(int argc, char* argv[ ])
{
    int a = 1, b;
    for(b = 1; b <= 10; b++)
    {
        if(a >= 8)
            break;
        if(a % 2 == 1)
        {
            a += 3;
            continue;
        }
        a -= 3;
    }
    printf("%d\n", b);
    return 0;
}
```

7. 下面程序的输出结果是\_\_\_\_\_

```
int main(int argc, char* argv[ ])
{
    const char* p = "12345";
    const char **q = &p;
    *q = "abcde";
    const char *s = ++p;
    p = "XYZWVU";
    printf("%c\n", **++s);
    return 0;
}
```

8. 以下代码的打印的结果是\_\_\_\_\_（假设运行在 32bit--x86 系列计算机上）

```
struct st_t
{
    int      status;
    short*   pdata;
    char     errstr[32];
};
struct st_t st[16];
char* p = (char*)(st[2].errstr + 32);
printf("%d", (p - (char*)(st)));
```

9. 以下程序的输出结果是\_\_\_\_\_。

```
#include <stdio.h>
int main(void)
{
    int i=0;
    while(1)    {
```

```

        printf("*");

        i++;

        if(i<3) break;
    }
    printf("\n");
}

```

A、 \*          B、 \*\*\*          C、 \*\*          D、 \*\*\*\*

10. 输出结果是\_\_\_\_\_

```

typedef struct{
    int a;
    int b;
} node;
int foo(int a[10], char b[10 ], node c, node d[10])
{
    printf("a = %d\n", sizeof a);
    printf("b = %d\n", sizeof b);
    printf("c = %d\n", sizeof c);
    printf("d = %d\n", sizeof d);
    return 0;
}

```

11. 求 sizeof(p)

```

char *p = "hello!";
char p[] = "hello!";
void *p = malloc(100);

```

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

12. 解释程序中 define 语句行，给出整个程序运行结果并说明理由（条件：Linux、32bits、GNU）。

```

#include <stdio.h>
struct A{
    short i;
    short j;
    char * ptr;
    long array[100];
    char b[4];
    char* c;
};
#define PRINT_ME (char *)&(((struct A *) 0)->c)
int main(int argc, char* argv[ ])
{
    printf("%d\n", PRINT_ME);
    return 0;
}

```

13. 写出下面程序执行结果\_\_\_\_\_（条件 Linux 32bits GNU）。

```

设 int arr[ ] = {6, 7, 8, 9, 10};

```

```

int *ptr      = arr;
*(ptr++)    += 123;
printf(" %d" ,*++ptr));

printf(" %d" ,*ptr++));

```

14. 内存布局问题 (stack,heap,data,bss), 写出下列所有定义的变量在内存中位置.

```

#include <stdio.h>
int  a = 0;
int  b;
extern int x;
static char  c, y = 'C';
int  main(int  argc, char* argv[ ])
{
    char      d=4;
    static short  e, f = 0x23;
    char      *p = (char *)malloc(20);
    printf("a=%d,  b=%d,  c=%d,  d=  %d,  e=%d",a,b,c,d,e);
    return 0;
}

stack:  _____
heap:   _____
data:   _____
bss :   _____

```

15. 请找出下面代码中的所有错误、写出错误原因、改正写在相应的行后。

说明：以下代码是把一个字符串倒序，如“abcd”倒序后变为“dcba”

0、#include "stdio.h"

1、#include "string.h" \_\_\_\_\_

2、main() \_\_\_\_\_

3、{

4、char\*src="hello,world"; \_\_\_\_\_

5、char\* dest=NULL; \_\_\_\_\_

6 int len=strlen(src); \_\_\_\_\_

7、dest=(char\*)malloc(len); \_\_\_\_\_

8、char\* d=dest; \_\_\_\_\_

9、char\* s=src[len]; \_\_\_\_\_

10、while(len--!=0) \_\_\_\_\_

11、d+=s--; \_\_\_\_\_

12、printf(dest); \_\_\_\_\_

13、return 0; \_\_\_\_\_

14、 }

16. 用变量 a 给出下面的声明, 答案写在其后。

- a) 一个整型数 (An integer)
- b) 一个指向整型数的指针 (A pointer to an integer)
- c) 一个指向指针的的指针, 它指向的指针是指向一个整型数
- d) 一个有 10 个整型数的数组 (An array of 10 integers)
- e) 一个有 10 个指针的数组, 该指针是指向一个整型数的
- f) 一个指向有 10 个整型数数组的指针 (A pointer to an array of 10 integers)
- g) 一个指向函数的指针, 该函数有一个整型参数并返回一个整型数
- h) 一个有 10 个指针的数组, 该指针指向一个函数, 函数有一个整型参数并返回一个整型数

17. 给出以下程序的运行结果并说明原因。

<pre>void GetMemory(char *p) {     p = (char *)malloc(100); } void Test(void) {     char *str = NULL;     GetMemory(str);     strcpy(str, "hello world");     printf(str); }</pre> <p>请问运行 Test 函数会有什么样的结果, 为什么?</p> <p>答:</p>	<pre>char *GetMemory(void) {     char p[] = "hello world";     return p; } void Test(void) {     char *str = NULL;     str = GetMemory();     printf(str); }</pre> <p>请问运行 Test 函数会有什么样的结果, 为什么?</p> <p>答:</p>
<pre>Void GetMemory2(char **p, int num) {     *p = (char *)malloc(num); } void Test(void) {     char *str = NULL;     GetMemory(&amp;str, 100);     strcpy(str, "hello");     printf(str); }</pre> <p>请问运行 Test 函数会有什么样的结果, 为什么?</p>	<pre>void Test(void) {     char *str = (char *) malloc(100); strcpy(str, "hello"); free(str); if(str != NULL) {     strcpy(str, "world");     printf(str); } }</pre> <p>请问运行 Test 函数会有什么样的结果, 为什么?</p>

什么? 答:	什么? 答:
-----------	-----------

18. 以下程序的运行结果是\_\_\_\_\_。

```
#include <stdio.h>
void abc(char *p)
{
    int i, j;
    for(i=j=0; *(p+i)!='\0'; i++)
        if(*(p+i) < 'n') {
            *(p+j) = *(p+i);
            j++;
        }
    *(p+j)='\0';
}
int main(void)
{
    char str[]="morning";
    abc(str);
    puts(str);
    return 0;
}
```

A、morig          B、 morning      C、 mig                  D、 or

19. 写出下列程序运行结果

```
#include <stdio.h>
int main(void)
{
    char a[10]={0,1,2,3,4,5,6,7,8,9};
    short *p=(short *) (a+2);
    short i=*p;
    printf("%x\n",i);
    return 0;
}
```

在小尾端上运行结果是多少?

20. 请问以下代码有什么问题?如何改正?

```
#include <stdio.h>
#include <string.h>
int main(void)
{
    char *str="Hehehe";
    strcpy(str,"hello");
    printf(str);
    return 0;
}
```