

每日一题C++方向day15_11月22日

一. 单选

1.

对两个字符a和b进行初始化:char a[]="ABCDEF";char b[]={ 'A','B','C','D','E','F'};则以下叙述正确的是 :

- ☐ A a数组比b数组长度长
- ☐ B a与b长度相同
- ☐ C a与b数组完全相同
- ☐ D a和b中都存放字符串

正确答案 : A

2. x是一个行列数均为1000二维数组 , 下面代码效率执行最高的是 ()

- ☐ A for(int j=0;j<1000;j++) for(int i=0;i<1000;i++) x[i][j]+=x[j][i];
- ☐ B for(int i=0;i<1000;i++) for(int j=0;j<1000;j++) x[i][j]+=x[j][i];
- ☐ C for(int i=0;i<1000;i++) for(int j=0;j<1000;j++) x[j][i]+=x[j][i];
- ☐ D for(int i=0;i<1000;i++) for(int j=0;j<1000;j++) x[i][j]+=x[i][j];

正确答案 : D

3. C++中关于堆和栈的说法, 哪个是错误的:

- ☐ A 堆的大小仅受操作系统的限制, 栈的大小一般一般较小
- ☐ B 在堆上频繁的调用new/delete容易产生内存碎片, 栈没有这个问题
- ☐ C 堆和栈都可以静态分配
- ☐ D 堆和栈都可以动态分配

正确答案 : C

4.

下面程序会输出什么 :

```
static int a=1;
void fun1(void){  a=2; }
void fun2(void){  int a=3; }
void fun3(void){  static int a=4; }
int main(int argc,char** args){
    printf("%d",a);
    fun1( );
    printf("%d",a);
}
```

```
fun2( );
printf("%d", a);
fun3( );
printf("%d",a);
}
```

- A 1 2 3 4
- B 1 2 2 2
- C 1 2 2 4
- D 1 1 1 4

正确答案：B

5.

In the main() function, after ModifyString(text) is called, what's the value of 'text'?

```
int FindSubString( char* pch )
{
    int count = 0;
    char *p1 = pch;
    while ( *p1 != '\0' )
    {
        if ( *p1 == p1[1] - 1 )
        {
            p1++;
            count++;
        }else {
            break;
        }
    }
    int count2 = count;
    while ( *p1 != '\0' )
    {
        if ( *p1 == p1[1] + 1 )
        {
            p1++;
            count2--;
        }else {
            break;
        }
    }
    if ( count2 == 0 )
        return(count);
    return(0);
}

void ModifyString( char* pText )
{
```

```
char * p1 = pText;
char * p2 = p1;
while ( *p1 != '\0' )
{
    int count = FindSubString( p1 );
    if ( count > 0 )
    {
        *p2++ = *p1;
        sprintf( p2, "%i", count );
        while ( *p2 != '\0' )
        {
            p2++;
        }
        p1 += count + count + 1;
    }else {
        *p2++ = *p1++;
    }
}
}
void main( void )
{
    char text[32] = "XYBCDCBABABA";
    ModifyString( text );
    printf( text );
}
```

- ☒ A XYBCDCBABABA
- ☐ B XYBCBCDA1BAA
- ☐ C XYBCDCBA1BAA
- ☐ D XYBCDDBA1BAB

正确答案：C

6.

所谓数据封装就是将一组数据和与这组数据有关操作组装在一起，形成一个集合，这集合也就是（ ）

- ☒ A 类
- ☐ B 对象
- ☐ C 函数体
- ☐ D 数据块

正确答案：A

7.

关于以下代码，哪个说法是正确的？

```
myClass::foo(){
    delete this;
}
..
void func(){
    myClass *a = new myClass();
    a->foo();
}
```

- ☐ A 它会引起栈溢出
- ☐ B 都不正确
- ☐ C 它不能编译
- ☐ D 它会引起段错误

正确答案：B

8.

假定CSomething是一个类，执行下面这些语句之后，内存里创建了____个CSomething对象。

```
CSomething a();
CSomething b(2);
CSomething c[3];
CSomething &ra = b;
CSomething d=b;
CSomething *pA = c;
CSomething *p = new CSomething(4);
```

- ☐ A 10
- ☐ B 9
- ☐ C 8
- ☐ D 7
- ☐ E 6
- ☐ F 5

正确答案：E

9.

下面这段代码运行时会出现什么问题？

```
class A
{
public:
    void f()
{
```

```

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

class B: public A
{
public:
    virtual void f()
    {
        printf("B\n");
    }
};

int main()
{
    A *a = new B;
    a->f();
    delete a;
    return 0;
}

```

- ☒ A 没有问题，输出B
- ☐ B 不符合预期的输出A
- ☐ C 程序不正确
- ☐ D 以上答案都不正确

正确答案：B

10.
下面这段代码会打印出什么？

```

class A
{
public:
    A()
    {
        printf("A ");
    }

    ~A()
    {
        printf("deA ");
    }
}

```

```
};

class B
{
public:
    B()
    {
        printf("B ");
    }
    ~B()
    {
        printf("deB ");
    }
};

class C: public A, public B
{
public:
    C()
    {
        printf("C ");
    }
    ~C()
    {
        printf("deC ");
    }
};

int main()
{
    A *a = new C();
    delete a;
    return 0;
}
```

- ☒ A A B C deA
- ☐ B C A B deA
- ☐ C A B C deC
- ☐ D C A B deC

正确答案：A

二. 编程

1. 标题：查找输入整数二进制中1的个数 | 时间限制：1秒 | 内存限制：32768K

请实现如下接口

```
public
```

```
static
```

```
int
```

```
findNumberOf1(
```

```
int
```

```
num)
```

```
{
```

```
/*  
请实现  
*/
```

```
return
```

```
0;
```

```
}
```

譬如：输入5，5的二进制为101，输出2

涉及知识点：

输入描述：

输入一个整数

输出描述：
计算整数二进制中1的个数

示例1:

输入
5

输出
2

正确答案：

2. 标题：手套 | 时间限制：3秒 | 内存限制：32768K | 语言限制：[Python, C++, C#, Java]

在地下室里放着 n 种颜色的手套，手套分左右手，但是每种颜色的左右手手套个数不一定相同。A先生现在要出门，所以他要去地下室选手套。但是昏暗的灯光让他无法分辨手套的颜色，只能分辨出左右手。所以他会多拿一些手套，然后选出一双颜色相同的左右手手套。现在的问题是，他至少要拿多少只手套(左手加右手)，才能保证一定能选出一双颜色相同的手套。

给定颜色种数 $n(1 \leq n \leq 13)$ ，同时给定两个长度为 n 的数组 $left, right$ ，分别代表每种颜色左右手手套的数量。数据保证左右的手套总数均不超过26，且一定存在至少一种合法方案。

测试样例：

4,[0,7,1,6],[1,5,0,6]

返回：10(解释：可以左手手套取2只，右手手套取8只)

输入描述：

输出描述：

示例1:

输入

输出

正确答案：