

The C Programming Language Exercise

关于作业：

各位小伙伴，协会第二次培训已经结束了。在接下来的学习中，你们也要继续坚持下去哦。赶快充充电，来做做练习，挑战挑战自己吧。

这是第二次培训的配套练习，希望大家能先尝试一下。如有任何疑问都可以在群里讨论或者联系群里的学长们。

请尽量在 11 月 10 之前写完，到时学长们要一行一行看你们的代码。关

于作业提交：写好后请尽量在 11 月 10 前以邮件附件的形式发送到

homework@vidar.club 源码编码规范必须采用 UTF-8。附件须为压缩包，压缩包命名格式“C2 你的名字”，如“C2 张三.zip”。该压缩包须包含每道题的 C 代码文件，命名格式“C2 题号名字”，如第 1 题“C2_01 张三.c”。

在代码文件中可以注明疑问和想法，学长们会抽时间将每份代码看过去并回复大家。

可能不是所有的小伙伴都能做出所有的题，没有关系，能够做出没有标明选做的题，你已经很厉害了。只要把会写的写就 OK，不会的空着没事的。但有一点，不要抄，尽量写。

可以参考网上的，但不要直接抄网上的，直接抄下来不去思考不去写，没有任何意义。

如有任何关于作业的问题，直接提出来就好。



1. 补全代码，使得 getElement 返回 a[b][c]的值

```
#include <stdio.h>
int getElement (int* a, int b, int c)
{
    // 在这补全代码
}
int main()
{
    int a[16][16] = {0};
    int b, c;
    scanf_s("%d %d", &b, &c);
    printf("%d\n", getElement((int *)a, b, c));
    return 0;
}
```

2. 内存对齐

在 x64 平台上，下面这份代码会输出什么呢？你能简单解释一下为什么吗？

```
#include <stdio.h>
#include <string.h>
#pragma pack(push)
#pragma pack(8)
typedef struct
{
    char a;
    unsigned long long b;
} aaa;
#pragma pack(pop)

int main()
{
    aaa ddd;
    printf("%ld %ld %ld\n", sizeof(ddd.a), sizeof(ddd.b), sizeof(ddd));
    return 0;
}
```



3. 指针、字符串

请实现一个 `mystrcmp` 函数，要求其声明如下：

```
int mystrcmp(const char* s1, const char* s2);
```

(假设 `s1` 与 `s2` 都保证指向合法的字符串，不存在指针值为 `NULL` 的情况)

4. 指针、const

现在，回头看看第三题，考虑这个问题：你书写的 `mystrcmp` 真的无法修改 `s1` 跟 `s2` 吗？试着在 `mystrcmp` 函数里里里修改 `s1` 与 `s2`，但是不改变其声明，并写下你对这道题（修改 `const` 的值）的理解

5. 内存排布

作为函数的参数时，`char []` 与 `char *` 这两个类型是等价的。但是在声明变量时却不等价，试阐述以下两种声明方式的区别：

```
int main()
{
    const char* str1 = "Hello, World!";
    char str2[] = "Hello, World!";
}
```



6. [附加题] C 语言也想有自己的“类”

补全横线，请不要动其它代码，使程序最后输出 23333

```
#include<stdio.h>

struct myClass
{
    int num;
    _____;
};

_____ printNum(_____)
{
    printf("%d\n", this.num);
}

int main()
{
    struct myClass test;
    test.num = 23333;
    test.func = printNum;
    test.func(test);
}
```

