

[C/C++ 基础知识] 面试再谈struct和union大小问题

原创 Eastmount 2015-09-23 03:54:48 4303 收藏 1

展开



Python+TensorFlow人工智能

该专栏为人工智能入门专栏，采用Python3和TensorFlow实现人工智能相关算法。前期介绍安装流程、基础语法、



Eastmount

¥9.90

订阅

最近找工作参加了很多笔试，其中考察结构体和联合体的大小问题是经常出现的一个问题。虽然题目简单而且分值比较低，但是还是想再给大家回顾下这些C和C++的基础知识。希望文章对你有所帮助~

PS：意外惊喜第三部分，所有权归它们公司所有。我只想分享学习并无它，望海涵~

一. 真题介绍

1.[2015-9 完美] 在IA32架构下，下面的union结构的sizeof大小为：_____

```
union PageLayout
{
    struct
    {
        int page_index;
        char key[5];
    };
    char dummy[10];
};
```

题解：

该题充分考察了结构struct和union联合的区别：

联合与结构的本质区别在于内存使用方式的不同。

结构中不同的成员使用不同的存储空间，一个结构所占的内存大小是结构中每个成员所占内存大小的总和，结构体中每个成员相互独立，是不能占用同一存储单元的。

联合大小取决于其中最大的数据类型内存分配大小，联合中内存是叠加存放的。

同一存储区域由不同类型的变量共享，这种数据类型就是联合(也称共同体)。

故：联合PageLayout中由struct和dummy[10]两部分组成，其中结构的大小为int型4字节+char8字节，因为它需要字节对齐为4的整数倍，结构12字节+dummy字符数组10字节。

答案：12

```
union PageLayout
{
    struct
    {
        int page_index;
        char key[5];
    };
    char dummy[10];
};
```

sizeof(PageLayout)
sizeof=12

2.[2013 完美世界] 求结构st的大小为_____。

```
struct st
{
    char ch,*ptr;
    union
```

```

{
    short a,b;
    unsigned int c:2,d:1;
};
bool f;
struct st *next;
};

```

题解:

该提醒主要考察结构体中嵌套联合的大小，原理同上。其中int型为4字节，short为2字节，指针相当于无符号的整形4字节，同时补齐则为实际大小。

char ch内存对齐后为4字节、char类型指针4字节、union为4字节（其中联合为最大数据类型的内存大小，short a 2字节、short b 2字节、无符号int4字节，位域问题见下）、bool型补齐4字节、struct类型指针4字节，总共占：4+4+4+4+4=20字节。

答案：20

位域：把一个字节中的二进制位划分为几个不同的区域，并说明每个区域的位数。

格式：类型说明符 位域名：位域长度，例如 unsigned int c:2表示c在内存中占2位

通过如下程序输出结构体中内存地址如下图所示：

```

int main()
{
    struct st student;
    printf("sizeof=%d\n", sizeof(st));
    printf("0x%p\n", &student.ch);
    printf("0x%p\n", &student.ptr);
    printf("0x%p\n", &student.a);
    printf("0x%p\n", &student.b);
    printf("0x%p\n", &student.f);
    printf("0x%p\n", &student.next);
    system("PAUSE");
    return 0;
}

```

Address	Field	Value	Size
0x0054FA00	ch	00 01 02 03	4
0x0054FA04	ptr	04 05 06 07	4
0x0054FA08	a	08 09 0A 0B	4
0x0054FA0C	b	0C 0D 0E 0F	4
0x0054FA10	f	10 11 12 13	4

4+4+4+4+4=20

3.[变形 完美] 求结构st的大小为_____.

```

struct st
{
    char ch,*ptr;
    union A
    {
        short a,b;
        unsigned int c:2,d:1;
    };
    bool f;
    struct st *next;
};

```

题解:

它与上题的区别联合声明A，表示定义的一个类型不用占用内存；而如果没有声明A，则表示声明了结构体中的一个成员，需要占内存。

内存大小为ch补齐4字节、char指针4字节、联合不占内存0字节、bool型补齐4字节、结构指针4字节，输出16字节。注意：如果联合后面添加ui，如 "union A{...}ui;" 此时输出结果为20。

答案：16

```
int main()
{
    struct st student;
    printf("sizeof=%d\n", sizeof(st));
    printf("0x%p\n", &student.ch);
    printf("0x%p\n", &student.ptr);
    printf("0x%p\n", &student.f);
    printf("0x%p\n", &student.next);
    system("PAUSE");
    return 0;
}
```

```
sizeof=16
0x0090FD6C
0x0090FD70
0x0090FD74
0x0090FD78
请按任意键继续. . .
```

二. 其他题型

代码分别如下所示:

```
struct A
{
    int page_index;
    char key[5];
};
```

输出: 12

其中int4字节+char两个4字节补齐

```
struct B
{
    char a;
    int b;
    double c;
};
```

输出16

其中补齐4字节+int型4字节+double8字节, 其中为什么补齐4而不是8呢? 思考下~

```
struct C
{
    char a;
    double c;
    char b;
};
```

输出: 24

输出补齐a8字节+double8字节+c补齐8字节

```
union
{
    long i;
    int k;
    char c;
    char s[4];
};
```

输出: 4

联合4字节, 其中由于联合各个成员使用共同的存储区域, 当向其中一个成员赋值时, 联合中的其他值也会发生变

化。

推荐文章: <http://blog.chinaunix.net/uid-26943148-id-3196468.html>

三. 其他经典考题

由于很多题目都要求不能泄露, 只能凭借记忆简单再分享几种常见的题型, 这些基础型题目是关于C\C++\数据结构的, 任何岗位都可能遇到。因为我个人报的岗位众多, 包括: C++开发、算法工程、NLP、PHP开发、大数据方向等, 但是还是建议:

"精>>杂 | 专一>>博爱 | LeetCode>>不做"

这么多笔试, 让我牢记一点: 山外有山, 人外有人, 尤其是程序猿, 自己真心太弱, 要学习的东西太多太多; 但是什么时候都不能丢失自我和自己感兴趣的东西, 即使再累再苦, 做自己喜欢的事情就是幸福, 比如写博客、玩爬虫、赏美文、学习新知识。即使半夜凌晨, 分享一篇博客或看到好的东西都让人欣喜, 这就是生活吧!

1.[2015-9 完美] `int n=0; while(n=1) n++; while`循环执行的次数是: ____.

答案: 无限循环

因为`while(n=1)`是个赋值语句, 表示动作始终为true

2.[2015-9 完美] 二叉树后序遍历序列为DEBFCA, 中序遍历序列为DBEAF, 则前序遍历顺序: ____.

提示: E代价也考察了类似题目

先序表示根->左->右、中序表示左->根->右、后序表示左->右->根。记住先序根在前面, 后序根在最后。

题中后序DEBFCA, 显然A为第一个根节点。前序最先输出A

3.[2015-9 完美] 下面程序的输出是多少: ____.

```
#define add(a+b) a+b
int main()
{
    printf("%d\n", 5 * add(3+4));
    return 0;
}
```

答案: 19

这是一道非常基础的考察宏定义的题目, 题目是错的, 应该改为`add(a,b)`。但显然不影响其功能, 它替换后结果为: `5*3+4=19`, 易错误的结果是输出35。显然它没有添加括号。再补充一道2015后端研发美团的类似题目。

4.[2015-9 美团] 多个源文件组成C程序, 经过编辑、预处理、编译、链接生成可执行程序, 下列哪个可以发现被调用的函数未定义?

答案: 链接

解析: 本题考查的是程序编译过程的基本知识。对于编译型程序设计语言C, 在程序编写完成后执行前, 主要进行预处理、翻译为目标代码和链接库函数等关键步骤。

在这三步中, 预处理分析程序中的宏定义并替换宏引用, 翻译主要针对一个编译单元(通常对应一个源文件)进行, 将该编译单元翻译为中间代码, 链接过程将各个编译单元中变量和函数的引用与其定义绑定, 确保程序中使用的所有变量和函数都存在对应实体。所以, 未定义的函数引用只能在链接过程中发现。

5.[2015-9 美团] 按入栈序列ABCDE, 不可能出栈的序列式: ____.

DECBA DCEBA ECDBA ABCDE

提示: 该题目完美、E代价等公司都有

答案: ECDBA

经典的考察出栈题目:

如果在草稿纸上画出入栈图就非常容易了。此时入栈ABCD

D => E

C => C

B => B

A => A

出D入E再出CBA，出DC入E再出BA，最后是入一个出一个，而E出后必须先D后C，故ECDBA错误。

6.二分查找第一轮查找的关键字是什么？快速排序第一轮结果是什么？堆排序第一轮后的结果是什么？(多为选择题)

7.[2015-9 360] Person类实例化new一个对象\$p，那如何使用对象\$p调用Person类中的方法：

A.\$p->getInfo() B.this->getInfo() C.\$p::getInfo() D.\$p=>getInfo()

提示：PHP方向

完美考了面向对象的继承，不能直接访问基类中继承的某个成员，通常是私有成员；内联函数、虚函数等知识。

8.[2015-9 360] 下列不是动态规划算法的基本要素？

A.马尔科夫性 B.建表填充 C.子问题叠代 D.最优子结构

9.[2015-9 360] 下列不要求最优子结构的：

A.分治法 B.贪心算法 C.动态规划 D.回溯法

提示：贪心算法和动态规划的共同点就是最优子结构。

10.[2015-9 360] TCP连接socket上调用recv函数，返回值为0表示：

- A.对端关闭连接
- B.连接错误
- C.对端发送长度为0数据
- D.还没收到对端数据

怀疑：A

TCP面向连接，保证数据安全、三次握手；UDP包可能丢失，但速度更快。(完美)

套接字编程send和recv参数也常考。recv的功能是从接收缓冲区读取(其实就是拷贝)指定长度的数据。recv返回的条件有两种：

(1). recv函数传入的应用层接收缓冲区已经读满

(2). 协议层接收到push字段为1的TCP报文，此时recv返回值为实际接收的数据长度

客户端的程序连接上服务器后recv函数阻塞接受，有时会返回0，说明接收超时服务器主动断开了连接,需要重新connect服务器。[参考](#)

E代价有道题目也比较好：

200-OK 请求成功

400-Bad Request 语义有无，当前请求无法被服务器理解

403-Forbidden 服务器已经理解请求，但拒绝执行它

404-Not Found 请求失败，请求资源未被服务器发现

500-Internal Server Error 服务器遇一个未曾预料的情况，导致无法完成请求处理

505-HTTP Version Not Support 服务器不支持或拒绝在请求中使用HTTP版本

11.数据库常考select语句、事务ACID

12.操作系统就是死锁及解决死锁方法、进程线程区别、阻塞执行就绪状态的天下

最后为什么说leetcode远远大于不做，因为最开始我挺反感A题的，后来实在是后悔A晚了。毕竟不是三年前疯狂刷题的我，同时很多题目都是看不出BUG的；尤其是笔试中很多Leetcode题目。包括：

完美的链表转置Reverse、E代价的判断单链表是否存在环（一步两步问题）、美团的循环 $n = n \& (n - 1)$ 计算n二进制中1的个数、360的求素数、阿里巴巴以对的形式判断最大堆结合二叉树、掌趣游戏二分查找、二叉树层次遍历(队列)、安全线性队列类等。

总之，希望文章对你有所帮助吧！还是低调点，都不敢发布出来。嘘~

(By:Eastmount 2015-9-23 凌晨4点 <http://blog.csdn.net/eastmount/>)



Eastmount   博客专家

原创文章 462 获赞 6725 访问量 525万+

关注

他的留言板