

2016届360公司PHP服务端开发笔试和面试之所得所感

原创 Eastmount 2015-10-17 02:11:05 10132 收藏

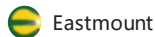
版权

分类专栏: [面试工作](#) [PHP基础知识及网站开发](#)



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、神经网络、可视化等，中间讲解CNN、RNN、LSTM等代码，后续复现图像处理...



¥9.90

[订阅博主](#)

这是一篇叙述自己在360公司参加笔试和面试的过程，可能面试的职位并不是你所学的方向，但是如果你能从中学到些什么或者吸取我的教训，那么作者就非常知足了。本着"学习别人是怎么失败的，活着出来的人才能成功"的目标，我从三个方面进行叙述：

第一部分：360公司笔试题

第二部分：面试过程

第三部分：注意事项及心得体会

同时，真心感谢360公司，我非常向往的一个公司。也非常感谢给我面试的那位大哥，让我真的学到了很多。所有题目版权归360所有，如果有不适的地方请告知我删除或修改。总之我认为：有的时候了解别人失败的案例比你总看别人成功的例子对你的帮助更大。看了下面我的本科毕设，你就会知道为什么我这么推崇这个公司了。

下载地址：<http://download.csdn.net/detail/eastmount/8591789>



一. 笔试题目

面试时间：2015年9月16日

面试岗位：PHP服务端开发工程师

职位要求：精通PHP/Python语言语法、掌握MySQL，基本了解Redis和MongoDB等各种DB、掌握HTML/CSS等。

题目难度：较难，选择题考得比较广，编程题一道极为简单一道比较复杂。

PS：因为当时题目都是一边做一边在草稿纸上抄的，可能有遗漏的地方，请海涵~

(一) 单选题：

1. MySQL存储过程的优点：

某选项可以多次调用、修改，网络负载降低

2. 找出/etc/my.conf文件属于哪个包（package），执行：

A.rpm -qf /etc/my.conf B.rpm -q /etc/my.conf
C.rpm -q | grep /etc/my.conf D.rpm -requires etc/my.conf

题解: 该题考察linux, 答案是A。其中-f Query package owning FILE, 赛马网:

- ivh: 安装显示安装进度--install--verbose--hash
- Uvh: 升级软件包--Update;
- qpl: 列出RPM软件包内的文件信息[Query Package list];
- qpi: 列出RPM软件包的描述信息[Query Package install package(s)];
- qf: 查找指定文件属于哪个RPM软件包[Query File];
- Va: 校验所有的RPM软件包, 查找丢失的文件[View Lost];
- e: 删除包

3.下面代码的运行结果:

- A.1 B.警告, 没定义a::\$myvar
C.2 D.一个错误, 没定义a::\$myvar

```
<?php
    class a{
        function a($x=1) {
            $this->myvar=$x;
        }
    }
    class b extends a{
        var $myvar;
        function b($x=2) {
            $this->myvar=$x;
            parent::a();
        }
    }
    $obj=new b;
    echo $obj->myvar;
?>
```

题解: 答案A. 参考

4.以下代码能正确显示图片的是:

PS: 由于4段代码, 仅仅手抄了B答案, 仅供参考

```
<?php
    header("content-type:image/jpeg");
    $img=imagecreatefromjpeg("images/scce.jpg").imagejpeg($img);
    imagedestroy($img);
?>
```

5.下面的脚本输出值为多少:

- A.5 B.2 C.10 D.NULL

```
<?php
    class my_class{
        var $value;
    }
    $a=new my_class;
    $a->my_value=5;
    $b=$a;
    $b->my_value=10;
```

```
echo $a->my_value; ?>
```

题解：通过 <http://www.mcqyy.com/RunCode/php/> 运行结果为C.10

\$b=\$a的赋值，仅仅是建立一个链接，赋值后对\$b的修改会影响\$a。

6. Person类实例化(new)一个对象\$p，那使用对象\$p调用Person类中getInfo方法：

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

题解：A，常用的考察PHP类定义调用方法的问题。

7. 下列关于工厂方法factory叙述正确的是那个：

8. $R=(A,B,C)$ 与SQL语句select distinct A from R where B=17等价关系代数表达式：

- A. $\pi_A(\sigma_{B=17}(R))$ B. $\sigma_{B=17}(\pi_A(R))$
C. $\sigma_{B=17}(\pi_A, C(R))$ D. $\pi_A, C(\sigma_{B=17}(R))$

题解：其中select语句对应 $\sigma_{B=17}(R)$ ，而Select distinct A为消除重复，答案为A。

投影操作 π 是一个关系操作，所谓的出现重复行是指多个记录在投影属性上具有相同的取值，例如参考百度百科：

学号	姓名	性别	年龄
01	艾伦	男	17
02	三笠	女	17
03	阿明	男	17

在性别和年龄两个属性上投影后数据集只保留这两个属性列，结果如下：

性别	年龄
男	17
女	17
男	17

其中第一行和第三行就是重复行，虽然来自不同记录，但是这两个属性上的内容相同。需要消除相同的行（SQL语句默认不消除重复），最后结果就是：

性别	年龄
男	17
女	17

9. 计算机cache，一主存层次采用相联映射方式，块大小为128字节，cache容量64块，按4块分组，主存容量为4096块，主存地址共需____位。

题解：由于主存容量为4096块，而每块为128个字，主存的总容量为512K字，故主存地址应为19位。主存地址应分为区号、组号、组内块号、块内地址号。可以看到，块内地址号应为7位，用以表示128个字。一组为4块，则组内块号用2位表示。Cache容量为64块共分16组，故组号需要用4位地址表示。剩余的即为区号，主存区号应为6位。|

10. 下列代码的运算结果为多少：

```
var a = new Array(2,3,4,5,6);
var sum=0;
for(i=1;i<a.length;i++)
sum+=a[i];
document.write(sum)
```

题解：计算 $3+4+5+6=18$

11. document对象的是：

- A. form B. link C. 三项都是 D. anchor

答案：C

12.mysql数据库还原命令是:

题解: 恢复、备份数据库备份数据库shell > `mysqldump -h host -u root -p`

13.求下列代码的时间复杂度, more than one answer is correct, choose the smallest one ().

```
for(i=0; i<n; i++)  
    for(j=1; j<=m; j*=2)  
        for(z=j/2; z<j; z++)
```

其中某项答案: $O(n \cdot \log(m) \cdot m)$

14.HTML5库抛弃了:

A.form B.applet C.frame D.center

题解: html5不再使用`fram`, 答案C。不再用frame、noframes和frameset, 这些标签对可用性产生负面影响。

HTML5中不支持frame框架, 只支持iframe框架, 或者用服务器方创建的由多个页面组成的符合页面的形式, 删除以上这三个标签。

15.HTML5中, input元素type属性默认值为:

A.search B.hidden C.text D.form

题解: 默认应该是text

16.下列代码的输出结果是:

A.24 B.17 C.72 D.36

```
d=lambda p:p*2  
t=lambda p:p*3  
x=2  
x=d(x)  
x=t(x)  
x=d(x)  
print x
```

题解: 感觉lambda表达式替换 $2*2=4$ $4*3=12$ $12*2=24$, 应该输出A。Right?

17.不是动态规划算法基本要素的是:

A.马尔可夫性 B.建表填 C.运用子项叠代 D.最优子结构

答案: A

18.不要求最优子结构的是:

A.分治法 B.贪心 C.动态规划 D.回溯

答案: D

19.下列叙述正确的是:

PS: 太长记不下来了

20.设有N堆沙子排成一排, 其编号为1,2,3,...,N($N \leq 100$)。每堆沙子有一定的数量。现要将N堆沙子并成一堆。归并的过程只能每次将相邻的两堆沙子堆成一堆, 这样经过N-1次归并后成为一堆。找出一种合理的归并方法, 使总的代价最小:

PS: 答案没记录下来, 但是这是典型的动态规划问题。

参考: <http://blog.csdn.net/abcjennifer/article/details/5805330>

21.下列不是分治法所能解决的问题特征是：

A.子问题的解无后效性

题解：只能怀疑答案是A，因为其它选项忘了，请见谅！分治法通常分为"分解-解决-合并"三个步骤，其中若干小规模的问题是可以解决的子问题，即具有最优子结构性质；最后将各个子问题的解合并为原问题的解。

22.<div>360</div>红色链接不正确的是：

A.a:link{color:red}

题解：该题考察网页基础知识，包括CSS定义超链接颜色等。A答案肯定正确，比较考察实际应用。

23.下列不属于结构性伪类的是：

A.E: root B.E: enabled C.E: first-child D.E: empty

24.TCP连接，socket调用recv函数返回值为0表示：

A.对端发送了一段长度为0的数据

B.对端关闭了连接

C.还没有收到对端数据

D.连接发生错误

题解：答案B。如果recv函数在等待协议接收数据时网络中断了，那么它返回0。默认 socket 是阻塞的。阻塞与非阻塞recv返回值没有区分，都是：

<0 出错 =0 连接关闭 >0 接收到数据大小。

25.需对文件进行随机存取，下列哪种文件物理结构不适合上述应用场景？

A.顺序文件 B.索引文件 C.链接文件 D.Hash文件

题解：答案C。链式存储结构的存储地址不一定连续，无法通过计算地址实现随机访问，只能顺序访问。如果要随机访问的话只能顺序查找，效率低下。

26.关于int *const ptr叙述正确的是：

A.ptr不可修改，*ptr可修改

B.ptr可以修改，*ptr可修改

C.ptr可以修改，*ptr不可修改

D.ptr不以修改，*ptr不可修改

题解：答案A。参考牛客网。const 的作用就是封锁它后面的东西，即后面的不可改变。

对于 **int *const ptr** 没有const关键字时为int* ptr，此时ptr是指向int的指针。加上const后，**const修饰并封锁 ptr**，即ptr的指向不可改变。

同理 **int const* ptr(等同 const int *ptr)**。const修饰 * 解引用，即指针指向的内容不可改变。

27.下列错误的用法是：

PS：其中某个答案为 D.typedef void (*FUN)()

28.下面代码fun(21)输出值多少：

```
int fun(int a) {  
    a^=(1<<5)-1;  
    return a;  
}
```

题解：首先(1<<5)表示左移5位，相当于1乘以2的5次方，即100000=32。

然后是异或运算：

21=010101 ^ 31=011111 =》 001010=10，结果为10。

29.sort的template正确的写法是:

- A. void sort(class A first, class A last, class B pred)
- B. void template(class A, class B) sort(A first, A last, B pred)
- C. template<class A> <class B> void sort(A first, A last, B pred)
- D. template<class A, class B> void sort(A first, A last, B pred)

题解: 参考[牛客网](#), 答案D。

函数模板的声明

模板函数格式是先声明模板类型, 然后才能使用。函数模板可以用来创建一个通用的函数, 以支持多种不同的形参, 避免重载函数的函数体重复设计。它的最大特点是把函数使用的数据类型作为参数。

函数模板的声明形式为:

```
template<typename 数据类型参数标识符>
<返回类型> <函数名> (参数表)
```

```
{
    函数体
}
```

格式 `template<class T1, class T2, ...> 返回值 函数名 (参数列表) { //函数体 }`

30.16位机器, 浪费多少空间?

```
struct {
    char a;
    int b;
    char a;
}
A.8    B.4    C.6    D.2
```

题解: 答案D。16位机器, char型占1字节, int型占2个字节。数据自动对齐, 实际结构体: 1 (char) + 1 (补齐) + 2 (int) + 1 (char) + 1 (补齐) = 6字节, 浪费2个字节空间。

参考我的博客: [\[C/C++基础知识\] 面试再谈struct和union大小问题](#)

31.一道关于A[m][n]数组处理的题目:

PS: 答案好像有644、676、696等。

32.关于C/C++宏定义错误的叙述是:

- A. 宏定义不检查参数正确性, 会有安全隐患
- B. 宏定义的常量更容易理解, 如果可以使用宏定义常量的话, 要避免使用const常量
- C. 宏的嵌套定义过多会影响程序的可读性, 而且很容易出错
- D. 相对于函数调用, 宏定义可以提高程序的运行效率

题解: 参考[牛客网](#), 答案B。参考: <http://bbs.csdn.net/topics/340089467>

使用const比使用define有以下几种好处:

- (1) const会进行数据类型检查, 而define不会
- (2) const存储在符号表\常量区, 表示值不能修改

33.下列值为多少: $1^2^{\dots^100}$

题解: 答案100, python代码如下所示:

```

>>>
>>> i=1
>>> s=0
>>> while i<=100:
>>>     s = s^i
>>>     print str(i),str(s)
>>>     i=i+1

```

By: Eastmount

1	1	91	0
2	3	92	92
3	0	93	1
4	4	94	95
5	1	95	0
6	7	96	96
7	0	97	1
8	8	98	99
9	1	99	0
10	11	100	100

4的倍数规律
1 i+1 0 i
所以当i=100时为原值

该题方法简单，简单计算1~8即可找到规律

34. 下列关于继承错误的是：

- A. 只能公有继承，不能私有继承
- B. 派生类可以访问基类protect成员
- C. 一个基类可以继承多个派生类，一个派生类可继承多个基类
- D. 基类中至少有一个虚函数可构成多态

35. 下列代码的输出值为多少：

```

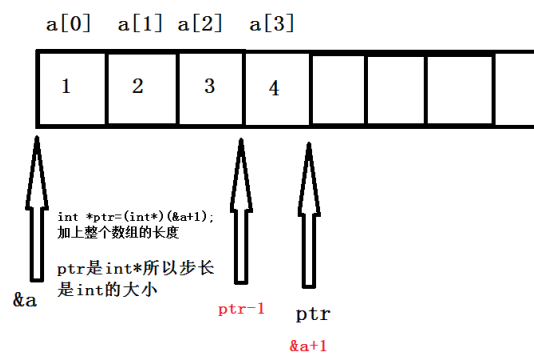
int main(int argc, char **argv)
{
    int a[4] = {1, 2, 3, 4};
    int *ptr = (int *)(&a + 1);
    printf("%d", *(ptr - 1));
}

```

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

题解：答案D。参考[赛马网](#)：

考察对于数组和指针的认识，指针加一的能力由类型决定。int*ptr=(int*)(&a+1); &a 和a 都指的是数组首元素的地址。不同的是a就是a+0，*(a+0) 就是a[0]，而&a+1相当于a[]数组类型的指针加1，此时指针加到数组的末尾。ptr接受后，由于Ptr的类型是int* 因此ptr-1即后退4字节。即指到最后一个元素。



36. 下列可作为对象继承之间的转换的是：

- A. static_cast
- B. dynamic_cast
- C. const_cast
- D. reinterpret_cast

题解：答案B。

dynamic_cast：在基类和派生类之间的转换，继承体系安全向下转型或跨系转型，找出某对象占用内存的起始点。static_cast：同旧式C转型，如int 到double。const_cast：常用于去除某个对象的常量性。reinterpret_cast

不具备移植性，常见用途是转化函数指针类型。

37.下列是获得实例化对象所属类名字的函数是：

- A.get_class_methods()
- B.get_class()
- C.get_classname()
- D.get_object_vars()

题解：答案B。PHP中没有get_classname()函数，其他如下：

- get_class — Returns the name of the class of an object
- get_object_vars — Gets the properties of the given object
- get_class_methods — Gets the class methods' names

(二) 编程题：

1.计算器的新功能

可视化程序设计一个新功能的计算器，输入一个数时，能将这个数分解为一个或多个素因子乘积的形式，并按素因子的大小排列显示出来，0-9这十个数字表示如下：每个数字占5*3大小的字符区域。

输入：多组测试数n(n<=1,000,000)

输出：每个数分成若干个素数乘积形式，从小到大输出。素因子之间用"*"形式连接。

例：

输入：

10
2

输出：

```
-      -  
|      |  
-  *  -  
|      |  
-      -  
  
-  
|  
-  
|  
-
```

首先需要计算素数组成，然后难点是怎样将数字一次性从上往下显示出来。

参考：<http://972459637-qq-com.iteye.com/blog/2244824>

2.研究生考试

政治100分，英语100分，数学150分，专业课150分。政治、英语要求单科不低于60分，数学、专业课要求单科不低于90分，总分不低于310分。总分350以上（含350）为公费，310-349分为自费。

请编程判断考生情况。

输入：正整数N，表示N组测试数据。每组4个正整数分：政治、英语、数学、专业

输出：Fail/Zifei/Gongfei

例：

3
61 62 100 120
80 80 120 100
55 90 130 130

输出:

Zifei

Gongfei

Fail

PS: 该题目比较简单, 基本为送分题。同时希望该部分题目对你有所帮助! 再次声明, 此部分为我一边做题一边抄在纸上, 所以有些遗漏的地方, 请原谅~

二. 面试过程

面试时间: 2015年10月9日

面试部门: 服务器端开发

面试地点: 360大厦

面试时长: 100多分钟

PS: 过程中可能存在一些遗漏的地方, 但是还是非常感谢那个面试的哥哥, 今天都还觉得给人很舒服的感觉。同时因为间隔时间太长, 最近也太忙, 不准备采用对话方式进行, 而是分几个步骤进行简单叙述。

第一部分 自我介绍

1. 首先简单问候面试官并递上自己的简历, 然后做个自我介绍;

2. 面试官通过我的简历, 让我介绍自己最拿得出手的项目, 我介绍的是知识图谱相关的项目, 包括: 传统搜索引擎的工作原理、知识图谱概念 (举例姚明身高、梁启超关系查询)、实体消歧与实体对齐、采用的VSM向量模型及聚类算法;

3. 面试官问我该阶段主要熟悉什么语言? 我说现在做得最多的是Python, 以前是C/C++, 当然Java、C#、PHP都做过, 毕竟语言都有通性, 但是想精通还是难。

4. 他说PHP比较简单, 他们是做底层服务器方向的, 今天的面试主要是问Unix相关知识; 我也赞同这个观点, 因为PHP可以通过一些开源框架实现, 同时也问了些自己做的WAMP网站。

第二部分 Unix为主

1. 面试官首先问我是否做过Unix相关的东西? 我说自己就简单做过Linux下的Python爬虫、脚本等。

2. 然后问了Unix下的网络编程会不会? 我简单介绍了Python的网络编程TCP\UDP的过程, 主要的三次过程如下, 同时Socket其他语言过程基本类似。

服务器:

```
ss = socket()           # 创建服务器套接字
ss.bind()                # 地址绑定到套接字上
ss.listen()              # 监听连接
inf_loop:                # 服务器无限循环
    cs = ss.accept()      # 接受客户端连接 阻塞式.程序连接之前处于挂起状态
    comm_loop:           # 通信循环
        cs.recv()/cs.send() # 对话 接受与发送数据
    cs.close()            # 关闭客户端套接字
ss.close()               # 关闭服务器套接字 (可选)
```

客户端:

```
cs = socket()            # 创建客户端套接字
cs.connect()             # 尝试连接服务器
```

```
comm_loop:           # 通讯循环
    cs.send()/cs.recv() # 对话 发送接受数据
cs.close()           # 关闭客户端套接字
```

3.然后他又问如果客户端出现异常，服务器怎样捕获这个异常呢？我当时想了下，提出了服务器可以设置一个时间点（心跳），当某段时间没有接受到该客户端的报文，则表示断开连接或异常错误。他又问我能不能把这段recv()函数写出来。我说不太会。

PS：回来后想了想，当时是不是在考察recv()函数的返回值：>0获得报文长度，=0客户端断开连接，<0连接发生异常错误。但确实自己也不会Unix下的网络编程。

4.面试官又问了些Unix下的fork相关的知识，我说没有接触过。还有些英文不知道是什么，自己英语太差了~

第三部分 算法和数据结构

1.面试官说：“你数据结构和算法应该很熟悉了吧！”我说：“还行，但是也忘记很多了。”自己确实很多基础知识都忘了很多，担心回答不上来。

2.面试官给我一张纸，有两段很长的代码（C语言），让我寻找两代码的区别。

这两段代码的主要区别就是参数一个是int，一个是double，当然前面还定义了些结构，代码里面的内容基本类似，相当于一个int型排序，一个double型排序。

他问我平时肯定会遇到这种情况，写两个函数代码过于冗余，怎样提炼成实现两种不同的类型排序，而且类型可以是float、结构体等等。

我说这有点类似于C++的模板啊！如果是C++就简单了，但是C语言主要是怎样判断这个类型呢？

PS：后来回来想了想，感觉类似于qsort快速排序的那种写法，通过const void *a实现，不知道是不是。但有同学怀疑是不是考察##的连接用法。

int型快排

```
int cmp1(const void *a, const void *b)
{
    return *(int*)a - *(int*)b;
}
```

```
qsort(num, len, sizeof(int), cmp1);
```

double型快排

```
int cmp(const void *a, const void *b)
{
    return *(double*)a > *(double*)b ? 1 : -1;
}
```

```
qsort(num, sum, sizeof(double), cmp);
```

char型快排

```
int cmp(const void *a, const void *b)
{
    return *(char*)a - *(char*)b;
}
qsort(str, sum, sizeof(char)*10, cmp);
```

3.上面代码没有写出来，那么你就做个最简单的吧！二叉树中序遍历非递归实现。

他又问我以前是怎么做的？我说通常都是三句话递归，这个题主要是考察通过栈模拟二叉树遍历递归的过程，然后写代码中。我失误了，栈写成队列了，然后队列是先进先出，又通过两个队列（一个输入队列、一个输出队列）模拟了一个栈实现了非递归遍历。

中序遍历：左孩子-根节点-右孩子，总体代码如下。参考

递归代码

```
void inOrder1(BinTree *root)    // 递归中序遍历
{
    if (root != NULL)
    {
```

```

        inOrder1(root->lchild);          cout<<root->data<<" ";
        inOrder1(root->rchild);
    }
}

```

非递归遍历

根据中序遍历的顺序，对于任一结点，优先访问其左孩子，而左孩子结点又可以看做一根结点，然后继续访问其左孩子结点，直到遇到左孩子结点为空的结点才进行访问，然后按相同的规则访问其右子树。因此其处理过程如下：

对于任一结点P，

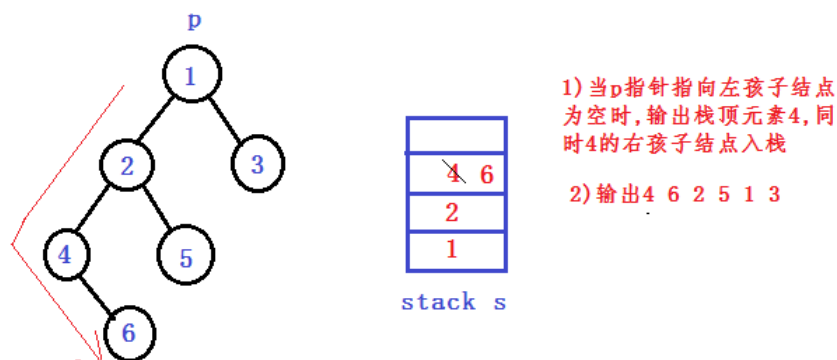
- 1)若其左孩子不为空，则将P入栈并将P的左孩子置为当前的P，然后对当前结点P再进行相同的处理；
- 2)若其左孩子为空，则取栈顶元素并进行出栈操作，访问该栈顶结点，然后将当前的P置为栈顶结点的右孩子；
- 3)直到P为NULL并且栈为空则遍历结束

```

void inOrder2(BinTree *root)    // 非递归中序遍历
{
    stack<BinTree*> s;
    BinTree *p=root;
    while(p!=NULL||!s.empty())
    {
        while(p!=NULL)
        {
            s.push(p);
            p=p->lchild;
        }
        if(!s.empty())
        {
            p=s.top();
            cout<<p->data<<" ";
            s.pop();
            p=p->rchild;
        }
    }
}

```

如下图所示：



第四部分 结束面试

1.面试官问我最近在看什么书？我说《Python核心编程》和《Web数据挖掘》，然后问我有没有看过Unix的书籍，我说看过《Unix编程艺术》前两章。他就给我推荐了三本书，希望我回去看这三本，其他可以放一边了，那些都是小儿科了。

《Unix高级环境编程》《TCP\IP协议》《Unix网络编程》

PS：这三本数都是Unxi传奇W.Richard Stevens的作品，当时以为推荐书，有面试了100分钟以为有戏。可惜

了，哈哈~

2.然后又问我有什么问题，我说想自己实现个小的搜索引擎系统；他给我分析了硬件设备、分词、索引、倒排序、Rank、推荐系统等等知识。

3.最后让我出去等了大概15分钟左右，好像他们那天也比较忙，最后还是方向不太对口被拒了。但我自己已经非常知足了，一方面从他那学到了很多，另一方面也深深认识到了自己的不足，当初随便报了个PHP方向居然能面试，我也不知道报了这个方向。

三. 注意事项及心得体会

在最后总结之间，说点题外话。在回学校之前，因为360公司就在798艺术工厂的旁边，我去到那里逛了3个多小时。写下这样一段话：

“今天早上来360面试，估计已跪，但仍不虚此行。会不会编程我不知道，但去了它旁边的798，发现自己还是有艺术细菌的。

好喜欢这种什么也不想，什么也不做，就静静地坐在角落，看着川流不息的人行的感觉。是那樣的踏实惬意，那样的无忧无虑~

因为不喜旅游，否则再忙也要出去看看这大千世界。但有时候又觉得做个井底之蛙也没有什么不好的，至少还可以每天看月亮。”

——Eastmount



最后简单总结下：

面试中经常考察的问题包括：

- 1.Socket套接字通信，TCP\UDP、同步异步解决方法；
- 2.基本算法和数据结构题目，包括二叉树遍历(含非递归)、快速排序(手写代码)、链表翻转等；
- 3.进程与线程的区别，是否写过线程相关的，如何解决同步、互斥等问题；
- 4.设计模式，如代理模式(图片浏览缩略图)、工厂模式、观察者模型等；
- 5.Cookie和Session的区别，缓存内容等，Android、前端开发常问；
- 6.后端开发常常会问Unix\Linux+C语言+网络通信的知识；
- 7.Python还会问爬虫、正则表达、开源框架Spider、docker、线程通信等；
- 8.自然语言处理会问分词、分类聚类、搜索引擎、推荐系统等；

提供几点建议：

- 1.申请职位一定要慎重考虑，一定是自己熟悉的东西，而不是各种职位都申请；
- 2.简历上的项目自己一定要熟悉，能说出项目的内容；
- 3.如果现在距离找工作早，建议尝试学习Linux、Unix下编程；
- 4.网络通信常常会问，简单的是TCP\UDP\Socket编程问题，三次握手等，如果深入就需要Unix相关知识；
- 5.算法很重要，如果有时间，一定要去做**LeetCode**题目，因为真的很多很多公司笔试面试题目都来自这里，比如链表翻转、字符串相似判断、二叉树遍历非递归等等；
- 6.如果有可能，尽量做些有深度的项目，简历中写"熟悉XXX语言、熟悉HTML或MySQL数据库"，显然不如"对Linux下网络编程比较熟悉，通过Spider或XX框架进行过分布式爬取"。尽量让自己的简历更加专业、有水平。
- 7.如果有开源项目或者研究过源码、驱动这些最好不过，但显然很少人做到。

最后希望文章对你有所帮助，如果有不足或错误的地方，还请海涵~希望大家都找到自己心仪的工作。

作者写这篇文章也不容易，最近真心太忙了，明天还有好几个笔试面试，自己也报了几个贵州的大学，还是想回去任教啊！一生的梦想，不知在何方；但是不论深处何地，做什么工作，都需要不断地学习，保持一颗平常心和健康的身體去生活。哈哈，加油~

(By:Eastmount 2015-10-17 深夜2点半 <http://blog.csdn.net/eastmount/>)